

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
ENG. DE CONTROLE E AUTOMAÇÃO

LEONARDO SANTANA REAL - 181909

**AUTOMAÇÃO DE AJUSTE DE
COMPONENTES DA MESA
TANGÍVEL E CRIAÇÃO DE
RECURSO EDUCACIONAL**

Porto Alegre
2024

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
ENG. DE CONTROLE E AUTOMAÇÃO

LEONARDO SANTANA REAL - 181909

**AUTOMAÇÃO DE AJUSTE DE
COMPONENTES DA MESA
TANGÍVEL E CRIAÇÃO DE
RECURSO EDUCACIONAL**

Trabalho de Conclusão de Curso (TCC-CCA)
apresentado à COMGRAD-CCA da Universi-
dade Federal do Rio Grande do Sul como parte
dos requisitos para a obtenção do título de *Ba-
charel em Eng. de Controle e Automação* .

ORIENTADOR:

Prof. Dr. Renato Ventura Bayan Henriques

CO-ORIENTADOR(A):

Prof. Dr. Rafael Antônio Comparsi Laranja

Porto Alegre
2024

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
ENG. DE CONTROLE E AUTOMAÇÃO

LEONARDO SANTANA REAL - 181909

**AUTOMAÇÃO DE AJUSTE DE
COMPONENTES DA MESA
TANGÍVEL E CRIAÇÃO DE
RECURSO EDUCACIONAL**

Este Trabalho de Conclusão de Curso foi julgado adequado para a obtenção dos créditos da Disciplina de TCC do curso *Eng. de Controle e Automação* e aprovado em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: _____

Prof. Dr. Renato Ventura Bayan Henriques, UFRGS
Doutor pela UFMG – Minas Gerais, Brasil

Banca Examinadora:

Prof. Dr. Renato Ventura Bayan Henriques, UFRGS
Doutor pela UFMG – Minas Gerais, Brasil

Prof. Dr. Heraldo José de Amorim, UFRGS
Doutor pela UFRGS – Porto Alegre, Brasil

Prof. Dr. Ana Paula Rigatti-Scherer, UFRGS
Doutora pela PUCRS – Porto Alegre, Brasil

Alceu Heinke Frigeri
Coordenador de Curso
Eng. de Controle e Automação

Porto Alegre, Fevereiro 2024

AGRADECIMENTOS

A Deus.

À minha base: minha família.

Ao orientador Renato, por idealizar e supervisionar este trabalho. Obrigado!

Ao co-orientador Laranja, por ajudar neste trabalho. Obrigado!

RESUMO

O presente estudo discorre sobre a automatização dos ajustes dos componentes de uma mesa tangível e a criação de um recurso educacional. Essa mesa é usada como instrumento para tratamento de crianças com Transtorno do Espectro Autista. Esse transtorno é caracterizado como um distúrbio do desenvolvimento neurológico que afeta a capacidade de comunicação social, interação social e comportamento de um indivíduo. A adoção de interfaces tangíveis com finalidade educacional configura-se como uma ferramenta que visa promover o desenvolvimento da linguagem, do raciocínio e do comportamento. Com o finalidade de automatizar a orientação do espelho e da câmera infravermelho, recursos como o Raspberry Pi Zero W, módulo PWM PCA9685 e o suporte PanTilt são empregados. Na elaboração do recurso educacional - que envolve a produção de um jogo com o propósito de acompanhar habilidades de raciocínio lógico, concentração e demais aspectos pertinentes - acessível através de um navegador, são aplicados recursos disponíveis no Eduba Editor.

Palavras-chave: Mesa Tangível, Transtorno do Espectro Autista, Raspberry Pi, Eduba Editor.

ABSTRACT

The present study addresses the automation of adjustments to the components of a tangible table and the creation of an educational resource. This table is used as a tool for the treatment of children with Autism Spectrum Disorder (ASD). This disorder is characterized as a neurological developmental disorder that affects an individual's ability to social communication, social interaction, and behavior. The adoption of tangible interfaces for educational purposes constitutes a tool aimed at promoting the development of language, reasoning, and behavior. In order to automate the orientation of the mirror and infrared camera, resources such as the Raspberry Pi Zero W, PWM PCA9685 module, and PanTilt support are employed. In the development of the educational resource - involving the creation of a game with the purpose of enhancing logical reasoning, concentration, and other relevant aspects - accessible through a web browser, resources available in the Eduba Editor are applied.

Keywords: Tangible Table, Autism Spectrum Disorder, Raspberry Pi, Eduba Editor.

LISTA DE ILUSTRAÇÕES

1	Marcadores fiduciais (reactIVision).	16
2	Mesa tangível utilizada.	21
3	Esquemático elétrico para automação da posição do espelho e da câmera infravermelho.	23
4	Esquemático elétrico do uso do módulo DTH11 e do Raspberry Pi Zero W para automação do sistema de ventilação.	25
5	Esquemático elétrico do uso dos <i>coolers</i> , do módulo relé de 8 canais, da fonte externa de 12 V e do Raspberry Pi Zero W para automação do sistema de ventilação.	25
6	Máquina de estados para demonstrar o funcionamento do programa (“Escolha de guia” como estado inicial).	26
7	Edição da tela inicial no Eduba Editor.	27
8	Demarcação de área de interação com o usuário em vermelho e área de <i>feedback</i> em azul.	28
9	<i>Layouts</i> disponíveis: com o nome dos guias (imagem da esquerda) e sem o nome dos guias (imagem da direita).	30
10	Sistema de ventilação para temperaturas inferiores a 20 °C (sistema desligado na imagem da esquerda e sistema ligado na imagem da direita).	30
11	Guias “Ajustar Infravermelho” (imagem da esquerda) e “Ajustar Espelho” (imagem da direita).	31
12	Guia “Ajuda”.	31
13	Sistema de ventilação: visão externa (imagem da esquerda) e visão interna (imagem da direita).	32
14	Câmera infravermelho com o suporte <i>Pan Tilt</i>	32
15	Na imagem superior, a estrutura encontra-se montada com o servo motor, conectada em dois pedaços de madeira para simular a caixa. Na imagem inferior, a estrutura está montada com o servo motor e o espelho.	33
16	Simulações no Eduba Editor.	34
17	Os cartões destinados ao jogo apresentam imagens de um lado para interação (imagem da esquerda) e fiduciais no outro (imagem da direita).	35
18	Raspberry Pi Zero W.	41
19	Servo motor SG90.	41
20	Módulo PWM PCA9685.	41
21	Suporte <i>Pan Tilt</i>	42
22	Módulo DTH11.	42
23	<i>Cooler</i> usado.	42
24	Módulo relé usado.	43

25	Ilustração da montagem de componentes para automação da posição do espelho e da câmera infravermelho.	44
26	Ilustração da montagem do módulo DHT11 e do Raspberry Pi Zero W para automação do sistema de ventilação.	45
27	Ilustração da montagem dos <i>coolers</i> , do módulo relé de 8 canais, da fonte externa de 12 V e do Raspberry Pi Zero W para automação do sistema de ventilação.	45
28	Capa do recurso educacional.	80
29	Três abordagens no recurso educacional.	80
30	Opções da trilha “Pareamento”.	81
31	Primeira tela da trilha “Pareamento: Animais”.	81
32	Segunda tela da trilha “Pareamento: Animais”.	81
33	Terceira tela da trilha “Pareamento: Animais”.	82
34	Quarta tela da trilha “Pareamento: Animais”.	82
35	Quinta tela da trilha “Pareamento: Animais”.	82
36	Primeira tela da trilha “Pareamento: Esportes”.	83
37	Segunda tela da trilha “Pareamento: Esportes”.	83
38	Terceira tela da trilha “Pareamento: Esportes”.	83
39	Quarta tela da trilha “Pareamento: Esportes”.	84
40	Quinta tela da trilha “Pareamento: Esportes”.	84
41	Opções da trilha “Cor”.	84
42	Primeira tela da trilha “Cor: Animais”.	85
43	Segunda tela da trilha “Cor: Animais”.	85
44	Terceira tela da trilha “Cor: Animais”.	85
45	Quarta tela da trilha “Cor: Animais”.	86
46	Quinta tela da trilha “Cor: Animais”.	86
47	Primeira tela da trilha “Cor: Objetos”.	86
48	Segunda tela da trilha “Cor: Objetos”.	87
49	Terceira tela da trilha “Cor: Objetos”.	87
50	Quarta tela da trilha “Cor: Objetos”.	87
51	Quinta tela da trilha “Cor: Objetos”.	88
52	Opções da trilha “Fonema”.	88
53	Primeira tela da apresentação da trilha “Fonema: /K/”.	88
54	Segunda tela da apresentação da trilha “Fonema: /K/”.	89
55	Terceira tela da apresentação da trilha “Fonema: /K/”.	89
56	Quarta tela da apresentação da trilha “Fonema: /K/”.	89
57	Quinta tela da apresentação da trilha “Fonema: /K/”.	90
58	Primeira tela da interação da trilha “Fonema: /K/”.	90
59	Segunda tela da interação da trilha “Fonema: /K/”.	90
60	Terceira tela da interação da trilha “Fonema: /K/”.	91
61	Quarta tela da interação da trilha “Fonema: /K/”.	91
62	Quinta tela da interação da trilha “Fonema: /K/”.	91
63	Primeira tela da apresentação da trilha “Fonema: /R/”.	92
64	Segunda tela da apresentação da trilha “Fonema: /R/”.	92
65	Terceira tela da apresentação da trilha “Fonema: /R/”.	92
66	Quarta tela da apresentação da trilha “Fonema: /R/”.	93
67	Quinta tela da apresentação da trilha “Fonema: /R/”.	93
68	Primeira tela da interação da trilha “Fonema: /R/”.	93

69	Segunda tela da interação da trilha “Fonema: /R/”	94
70	Terceira tela da interação da trilha “Fonema: /R/”	94
71	Quarta tela da interação da trilha “Fonema: /R/”	94
72	Quinta tela da interação da trilha “Fonema: /R/”	95
73	Tela de felicitações do recurso educacional	95

LISTA DE TABELAS

1	Componentes usados para montagem da mesa tangível.	21
---	---	----

LISTA DE ABREVIATURAS

ASD	<i>Autism Spectrum Disorder</i>
CSS	<i>Cascading Style Sheets</i>
DI	<i>Diffused Illumination</i>
GPIO	<i>General Purpose Input/Output</i>
GPL	<i>General Public License</i>
HTML	<i>HyperText Markup Language</i>
I2C	<i>InterIntegrated Circuit</i>
IoT	Internet das Coisas
IR	Iluminação Infravermelha
LEDs	<i>Light Emitting Diodes</i>
MDF	<i>Medium Density Fiberboard</i>
PHP	<i>Hypertext Preprocessor</i>
PVC	<i>Polyvinyl Chloride</i>
PWM	<i>Pulse Width Modulation</i>
SDA	<i>Serial Data</i>
SCL	<i>Serial Clock</i>
TEA	Transtorno do Espectro Autista
TUIs	Interfaces Tangíveis de Usuário

SUMÁRIO

1	INTRODUÇÃO	11
2	REVISÃO DA LITERATURA	12
2.1	Transtorno do Espectro Autista	12
2.2	Automação	14
2.3	Mesa Tangível	15
2.4	Python	16
2.5	Desenvolvimento <i>Web</i>	17
3	METODOLOGIA	19
3.1	Ajustes	19
3.1.1	Projeto Base	19
3.1.2	Raspberry Pi Zero W	22
3.1.3	Implementação	22
3.1.3.1	Posicionamento do Espelho e da Câmera Infravermelho	22
3.1.3.2	Sistema de Ventilação	24
3.1.3.3	Desenvolvimento em Python	25
3.2	Recurso Educacional	26
3.2.1	Eduba Editor e Nidaba Player	26
3.2.2	Elaboração	27
4	RESULTADOS	29
4.0.1	Programa e Componentes	29
4.0.2	Jogo da Estrela	33
5	CONCLUSÕES	36
	REFERÊNCIAS	37
	APÊNDICE A - COMPONENTES UTILIZADOS	41
	APÊNDICE B - ILUSTRAÇÃO DA MONTAGEM	44
	APÊNDICE C - CÓDIGO ELABORADO EM PYTHON	46
	APÊNDICE D - INSTRUMENTO EDUCACIONAL	80

1 INTRODUÇÃO

Cresce de maneira substancial a demanda por recursos tecnológicos destinados à aplicação na educação inclusiva, especialmente por ferramentas que possam desempenhar dupla função: a de tecnologia assistiva, capaz de superar obstáculos para estudantes que manifestam algum tipo de deficiência, e simultaneamente como um recurso interativo e inovador para todos os envolvidos. Nesse contexto, insere-se a proposta inovadora de uma plataforma digital voltada à produção de recursos educacionais inclusivos, baseados em mesa tangível. Tal abordagem visa possibilitar ao próprio educador a criação de aplicações conforme as demandas individuais de cada aluno. A incorporação de recursos provenientes da tecnologia da comunicação e informação em contextos de ensino-aprendizagem oferecem condições para envolver as crianças, fomentar a interação e a concepção do conhecimento através da comunicação textual, visual, motora, da comunicação (das crianças com os profissionais envolvidos) e com os objetos de aprendizagem (PASSERINO; SANTAROSA, 2002).

Com o propósito de aprimorar um recurso de inclusão na sociedade, o presente estudo tem como objetivo a implementação da automatização no ajuste de componentes de uma mesa tangível utilizada como ferramenta no tratamento de crianças com Transtorno do Espectro Autista (TEA). Adicionalmente, almeja-se a criação de um jogo destinado a ser empregado como recurso educacional.

Inicialmente, são abordados os temas que estabelecem o contexto e os conceitos relevantes para o desenvolvimento desta pesquisa. Subsequentemente, são expostos os recursos empregados com vistas à consecução do mencionado propósito. O presente estudo constitui a fase conclusiva do Trabalho de Conclusão de Curso, englobando, portanto, a análise final.

A educação especial inclusiva se configura como uma abordagem pedagógica que mescla elementos do ensino regular com os princípios da educação especial. Mediante essa metodologia, propicia-se a integração entre crianças que apresentam distintas necessidades educacionais, por meio da adoção e implementação de métodos que possam ser empregados tanto por estudantes com algum tipo de deficiência quanto por aqueles que não a possuem. A inclusão, neste contexto, emerge como um reconhecimento da pluralidade de formas de aprendizado, voltado para a exploração de alternativas que propiciem uma aprendizagem multifacetada para todos os discentes, independentemente da presença ou ausência de deficiência (PREUSS, 2021).

2 REVISÃO DA LITERATURA

Este capítulo demonstra conceitos que fundamentam o desenvolvimento deste trabalho. Inicialmente, são fornecidas informações sobre o Transtorno do Espectro Autista, seguidas pela exposição sobre a automação, a mesa tangível, a linguagem de programação Python e o desenvolvimento *web*.

2.1 TRANSTORNO DO ESPECTRO AUTISTA

O Transtorno do Espectro Autista (TEA) é caracterizado como um transtorno do desenvolvimento neurológico que compromete a comunicação social, a interação social e o comportamento de um indivíduo. Sua causa ainda é desconhecida, mas acredita-se que uma combinação de fatores genéticos e ambientais contribua para o seu surgimento (ASSOCIATION, 2013).

Os primeiros sinais desse transtorno podem ser identificados ainda na infância, nos primeiros anos de vida. Os sintomas variam de pessoa para pessoa, mas comumente incluem dificuldades em estabelecer contato visual, linguagem deficitária, comportamentos repetitivos, resistência a mudanças na rotina e dificuldades em lidar com estímulos sensoriais (ROSA et al., 2022).

Uma das características mais distintas do Transtorno do Espectro Autista é a dificuldade na interação social, onde indivíduos têm dificuldade em compreender e expressar emoções, manter conversas e interpretar pistas sociais sutis presentes nas interações diárias. Esse desafio pode levar ao isolamento social e dificuldades em estabelecer relações significativas (BAUMINGER; KASARI, 2000).

É comum observar que indivíduos no espectro autista manifestam interesses restritos e intensos em certos temas. Essa característica pode influenciar desfavoravelmente as atividades cotidianas e a capacidade de adaptação dessas pessoas (TIAN; GAO; YANG, 2022). Comportamentos estereotipados ou repetitivos englobam estereotipias motoras simples, tais como o movimento de abanar as mãos e estalar os dedos, bem como a utilização recorrente de objetos, como o ato de girar moedas e alinhar itens, além da manifestação repetitiva de fala. A adesão excessiva a rotinas e os padrões restritos de comportamento podem ser expressos por meio da resistência a mudanças. Este fenômeno engloba o sentimento de desconforto diante de alterações aparentemente mínimas, a exemplo da substituição da embalagem de um alimento predileto. Além disso, tais padrões podem se refletir em comportamentos ritualizados, tanto verbais quanto não verbais, tais como a repetição constante de questionamentos e a prática de percorrer um espaço físico específico. Interesses altamente limitados e fixos nesse transtorno tendem a demonstrar anormalidade em sua intensidade ou foco (PSIQUIATRIA, s.d.).

Embora o TEA seja um transtorno complexo e desafiador, é importante reconhecer que cada indivíduo é único, com habilidades e desafios específicos. Alguns podem exibir

habilidades excepcionais em áreas como matemática, música ou memorização, conhecidas como Savantismo. O Savantismo é um fenômeno caracterizado pelo surgimento de habilidades excepcionais em indivíduos com *déficits* significativos em outras áreas. Essas habilidades extraordinárias podem se manifestar em campos como matemática, música, arte ou memorização. Ainda não se detém conhecimento integral acerca dos fatores envolvidos na ocorrência do Savantismo, no entanto, estudos indicam a possibilidade de que determinadas regiões cerebrais estejam relacionadas com esse fenômeno. Essas habilidades únicas podem trazer tanto desafios quanto oportunidades para a vida desses indivíduos, e sua compreensão é fundamental para o desenvolvimento de estratégias de apoio adequadas (TREFERT, 2014). A mesa tangível pode ser um instrumento no diagnóstico dessa síndrome.

O diagnóstico do Transtorno do Espectro Autista é realizado por profissionais de saúde qualificados, tais como psicólogos, psiquiatras e neuropediatras, por meio de observação cuidadosa do comportamento e entrevistas com pais ou cuidadores. O diagnóstico precoce é fundamental para possibilitar o início de intervenções adequadas o quanto antes (JOHNSON; MYERS, 2007).

O tratamento do TEA requer uma abordagem multidisciplinar, envolvendo terapia comportamental, terapia ocupacional, fonoaudiologia e apoio educacional (REICHOW et al., 2012). O objetivo é auxiliar a pessoa a melhorar suas habilidades de comunicação, interação social e reduzir comportamentos típicos. Não há uma cura para esse transtorno, mas com o suporte adequado, as pessoas nessa situação podem aprender a lidar com suas dificuldades e desenvolver seu potencial máximo (HOWLIN, 2003).

É essencial que as escolas ofereçam suporte educacional especializado e que os ambientes de trabalho sejam adaptados para atender às necessidades das pessoas com TEA (BARON-COHEN; LOMBARDO, 2017). O diagnóstico precoce e intervenções adequadas são fundamentais para ajudar a pessoa a alcançar seu potencial máximo. A sociedade desempenha um papel crucial: uma comunidade inclusiva, compreensiva e adaptada às necessidades das pessoas com TEA é essencial para promover a plena inclusão em todas as áreas da vida (WORLD... , 2011).

As histórias sociais são intervenções baseadas em narrativas que ajudam crianças com TEA a melhorar sua compreensão de situações sociais. Essas histórias podem retratar vários cenários, como saudações, turnos de fala e compartilhamento, que são desafiadores para esse público. Ao usar suportes visuais e linguagem clara, as histórias sociais fornecem uma estrutura apropriada para que as crianças entendam e respondam a pistas sociais (GRAY, 2010).

Outro recurso para pessoas com TEA é o uso de suportes visuais. Indivíduos com Transtorno do Espectro Autista frequentemente se destacam no processamento visual, tornando os recursos visuais eficazes na promoção da aprendizagem e comunicação. Esses suportes podem incluir gráficos, cronogramas e imagens usadas para representar tarefas e regras. Esses recursos fornecem uma estrutura concreta e visual à qual as crianças com TEA podem se referir, reduzindo a ansiedade e promovendo a independência. Por exemplo, um “cronograma visual” pode representar visualmente a sequência de atividades durante o dia escolar, ajudando essas crianças a seguir rotinas e transições de forma mais suave (HODGETTS; ZWAIGENBAUM; NICHOLAS, 2017).

Além das dificuldades de comunicação social, algumas pessoas com TEA podem experimentar sensibilidades sensoriais. Elas podem ser hipersensíveis ou hipo sensíveis a certos estímulos, como toque, som ou luz. A manifestação desse interesse ocorre por meio de reações extremadas a sons e texturas específicas, além do comportamento de cheirar ou

tocar objetos de maneira excessiva. Outros sinais incluem um encantamento notável por luzes ou objetos giratórios e, ocasionalmente, uma aparente indiferença a sensações de dor, calor ou frio. Reações extremas ou rituais que abarcam o gosto, cheiro, textura ou aparência da comida, assim como restrições alimentares excessivas, são frequentemente observados no contexto do Transtorno do Espectro Autista. Recursos focados em sensorialidade podem ajudar os indivíduos com TEA a gerenciar e regular suas experiências sensoriais. Por exemplo, pausas sensoriais usando ferramentas sensoriais como brinquedos de aperto, cobertores pesados ou fones de ouvido com cancelamento de ruído podem ajudar as crianças com TEA a manter o foco e a regulação emocional durante tarefas desafiadoras (SOUZA AKAHOSI FERNANDES; POLLI; AGNELLI MARTINEZ, 2021). Esses recursos criam um ambiente amigável aos sentidos, facilitando a aprendizagem e reduzindo a sobrecarga sensorial.

Com o crescente uso da tecnologia na educação, diversos aplicativos e softwares foram desenvolvidos para assistir pessoas com TEA em sua jornada de aprendizado. Por exemplo, aplicativos de *tablets* oferecem experiências interativas e envolventes que podem ser adaptadas às necessidades específicas de crianças com TEA (RAMDOSS et al., 2011). Esses aplicativos oferecem dicas visuais, *feedback* adaptativo e experiências de aprendizagem personalizadas, aumentando o engajamento e motivação desse público.

2.2 AUTOMAÇÃO

A automação é um conceito amplamente utilizado na atualidade, em diferentes setores. Basicamente, consiste na utilização de diferentes tecnologias para tornar os processos de produção mais eficientes, rápidos e precisos. Um aspecto específico da automação é o ajuste de posição de equipamentos, que tem sido amplamente explorado para otimizar os processos (JANEKOVÁ; FABIANOVÁ; KÁDÁROVÁ, 2023).

A história da automação remonta aos primórdios da Revolução Industrial, no século XVIII, quando as indústrias começaram a utilizar máquinas movidas a vapor e água, substituindo o trabalho humano em diversas tarefas (ROSA, 2020). Com a invenção das máquinas a vapor e o desenvolvimento da indústria têxtil, fábricas passaram a utilizar máquinas para executar tarefas anteriormente realizadas manualmente. Isso permitiu um aumento significativo na produção e a redução do tempo necessário para fabricar produtos (HOBSBAWM, 2018). Entretanto, foi na década de 1940 que a automação começou a ganhar força, com o desenvolvimento dos primeiros sistemas de controle automatizados (IMD, s.d.). Inicialmente, esses sistemas eram bastante simples, utilizando relés eletromecânicos como interruptores eletromagnéticos programáveis. Na metade do século XX, a automação de ajustes de equipamentos passou a ser destacada na indústria. Com o avanço da eletrônica e da computação, surgiram sistemas automatizados de controle capazes de ajustar os equipamentos automaticamente, mantendo a produção dentro de parâmetros pré-definidos. Esses sistemas, conhecidos como controle automático, possibilitaram maior precisão e eficiência na produção industrial (OGATA, 2016). Com o avanço da eletrônica, a automação de ajustes de equipamentos se expandiu para outros setores, como a indústria automobilística, onde robôs foram incorporados às linhas de produção para executar tarefas repetitivas e perigosas. Isso resultou em maior produtividade, menor desperdício e maior segurança para os trabalhadores (FREEMAN; LOUCA, 2001).

Atualmente, os sistemas de controle automatizados para ajustes de posição de equipamentos são compostos por diferentes elementos, como sensores, atuadores e contro-

ladores. Os sensores são responsáveis por captar informações do ambiente, como posição, velocidade e temperatura, e enviá-las para o controlador. Os atuadores, por sua vez, são dispositivos que realizam movimentos, a partir das informações recebidas do controlador. A automação continua avançando com a chegada da inteligência artificial e da robótica avançada. Atualmente, a automação é aplicada em diversas áreas, como agricultura, saúde, transporte e serviços (HIBRIDA, s.d.).

A automação foi responsável por uma revolução na configuração de componentes de equipamentos, proporcionando uma variedade de vantagens para fabricantes e operadores. Na automação, é possível que o usuário controle facilmente a posição dos equipamentos por meio de comandos via *smartphone* ou voz. O processo de configuração de componentes de equipamentos é simplificado e acelerado pela automação, convertendo-o de uma tarefa intensiva em mão de obra para uma operação rápida e precisa. Ao utilizar algoritmos avançados e mecanismos controlados por computador, a automação permite o ajuste contínuo das posições dos componentes de equipamentos. Essa abordagem diminui a probabilidade de ocorrência de erros humanos (VUALA; SOUSA; CARTH, s.d.). Essa redução contribui para o aprimoramento do desempenho global do equipamento e para a minimização de eventuais períodos de inatividade e custos de manutenção associados a configurações inadequadas.

2.3 MESA TANGÍVEL

Uma mesa tangível, também denominada mesa interativa, caracteriza-se como uma superfície plana equipada com sensores, atuadores e tecnologias de exibição que possibilitam a interação direta com conteúdo digital e objetos físicos. Projetadas para estreitar a lacuna entre o mundo físico e virtual, essas mesas proporcionam uma experiência multissensorial e imersiva aos usuários. Os três principais componentes de uma mesa tangível são a superfície da mesa, a tecnologia de detecção e a tecnologia de exibição. A superfície da mesa, geralmente confeccionada em materiais como vidro ou acrílico, destacando-se por sua excelente transparência e durabilidade, atua como interface física entre o usuário e o sistema. A tecnologia de detecção, incorporando câmeras e sensores, registra as ações do usuário na superfície da mesa. A tecnologia de exibição, por sua vez, utiliza projetores ou telas de LCD embutidas sob a superfície da mesa para projetar conteúdo digital (PREUSS, 2021).

A mesa tangível oferece uma ampla gama de funcionalidades, tornando-se uma ferramenta versátil para diversas aplicações. Sua capacidade primária envolve a manipulação de objetos, permitindo que usuários interajam fisicamente com elementos virtuais na superfície da mesa, proporcionando uma manipulação e controle intuitivos. Essa característica encontra aplicações em áreas como prototipagem virtual, arquitetura e design industrial.

No âmbito educacional, as mesas tangíveis transformaram a dinâmica de interação dos alunos com o conteúdo de aprendizado. Sua natureza tátil permite a exploração prática de conceitos, tornando o aprendizado mais envolvente e interativo (GLUZ et al., 2018). Além disso, as mesas tangíveis facilitam a colaboração entre os alunos por meio da interação *multitouch*, permitindo que trabalhem juntos em projetos ou resolvam problemas coletivamente. O uso de objetos sobre superfícies tangíveis amplia as oportunidades de crescimento para as crianças, proporcionando benefícios para o desenvolvimento da motricidade fina e ampla, aprimorando o sentido do tato e promovendo uma conexão mais efetiva entre o mundo concreto e o virtual (CARBAJAL, 2016).

Na engenharia mecânica, as mesas tangíveis podem aprimorar o processo de prototipagem. Engenheiros podem interagir diretamente com modelos virtuais, realizando modificações e avaliando seus designs em tempo real. Isso contribui para a melhoria do ciclo de desenvolvimento do produto e a redução do tempo de lançamento no mercado. Na área da saúde, as mesas tangíveis têm se mostrado ferramentas valiosas para profissionais médicos. A integração da tecnologia de imagem médica às mesas tangíveis permite que médicos visualizem e manipulem dados médicos de maneira mais interativa e imersiva. Esse avanço pode auxiliar no diagnóstico, no planejamento cirúrgico e na educação do paciente, resultando em melhores resultados médicos (LEBRUN et al., 2022).

A capacidade de tocar e manipular objetos em uma mesa proporciona uma experiência tangível que aprimora o engajamento do usuário. Um desafio enfrentado reside no custo de implementação das mesas tangíveis, que pode ser elevado para o desenvolvimento e manutenção. Além disso, o tamanho e o peso das mesas tangíveis podem limitar sua acessibilidade e portabilidade.

O rastreamento dos objetos na mesa tangível fundamenta-se em identificação visual, a qual se dissipa quando o objeto é removido do contato com a superfície da mesa. Identificadores únicos, conhecidos como fiduciais, são incorporados à base dos objetos, viabilizando uma leitura eficiente (COSTANZA; ROBINSON, 2003).

Na mesa tangível, é possível a utilização de objetos do cotidiano e das funções multitoque. O reconhecimento da superfície tangível ocorre por meio de uma câmera com iluminação IR (Iluminação Infravermelha) projetada para refletir a ponta dos dedos e marcadores impressos. A Iluminação Difusa ou *Diffused Illumination* (DI) é empregada para iluminar os fiduciais.

Para aprimorar a utilização desses dois sistemas, é possível empregar uma superfície acrílica que favoreça a refração interna frustrada da luz infravermelha (IR) e permita a emissão de luz fora da superfície. O *framework* conhecido como reactTIVision, voltado para o reconhecimento visual de dedos e fiduciais, realiza a análise da imagem capturada por uma câmera conectada ao sistema. Este *framework* utiliza o padrão TUIO (KALTENBRUNNER, 2009), transmitindo informações em pacotes *User Datagram Protocol* para o software responsável pela aplicação e a saída da imagem da mesa.

Os fiduciais (Figura 1) possuem um padrão com o objetivo de proporcionar um rastreamento mais sólido e rápido dos objetos, facilitando a ação do reactTIVision, que informa a posição e a orientação do fiducial na mesa. O reactTIVision é de distribuição livre.

Figura 1: Marcadores fiduciais (reactTIVision).



Fonte: Kaltenbrunner (2009)

2.4 PYTHON

O Python é uma linguagem interpretada, de alto nível e conhecida por sua simplicidade e legibilidade. Seu uso é difundido em vários domínios, abrangendo desde o

desenvolvimento web até aplicações em ciência de dados, inteligência artificial e desenvolvimento de aplicativos móveis.

A principal vantagem do Python reside no amplo suporte a bibliotecas e *frameworks*. Bibliotecas renomadas, como NumPy, Pandas, TensorFlow e Django, disponibilizam ferramentas e funcionalidades poderosas para áreas específicas, como computação científica, manipulação de dados, aprendizado de máquina e desenvolvimento web, respectivamente. A diversidade de bibliotecas e pacotes, como SciPy, oferece soluções prontas para uso em cálculos numéricos complexos, análise de dados e visualização, permitindo a reutilização de soluções existentes na abordagem de novos problemas (BORGES, 2010).

O suporte do Python à programação orientada a objetos é robusto e intuitivo. O princípio dessa abordagem permite que desenvolvedores criem código modular e fácil de manter, encapsulando dados e comportamentos dentro de objetos. Adicionalmente, recursos como decoradores e geradores enriquecem a funcionalidade e flexibilidade do código Python. Decoradores possibilitam a modificação do comportamento de funções ou classes sem a necessidade de alteração direta no código-fonte. Já os geradores facilitam a geração eficiente de grandes sequências de valores, economizando recursos de memória (AWARI, s.d.). Ferramentas de perfil, como cProfile, são aplicáveis para identificar áreas do código que consomem recursos excessivos, permitindo otimizações eficientes (FOUNDATION, s.d.).

As capacidades de integração do Python com outras linguagens de programação, como C/C++, conferem-lhe um caráter valioso para desenvolvedores envolvidos em projetos que demandam tanto *scripts* de alto nível quanto otimizações de baixo nível. Essa versatilidade possibilita a combinação de vantagens oferecidas por diferentes linguagens de programação, resultando em ganhos de produtividade. A natureza *open-source* do Python promove a constante atualização e aprimoramento da linguagem e suas bibliotecas, impulsionados por uma comunidade global de desenvolvedores (ROCHA, s.d.).

2.5 DESENVOLVIMENTO WEB

Diante do progresso tecnológico e da expansão da internet, nota-se um crescente incremento na demanda por *websites* e aplicações *web* que possuam características interativas. A relevância dos *websites* responsivos, isto é, capazes de se adaptar a distintos dispositivos e dimensões de tela, assume um papel crucial na oferta de uma experiência de navegação satisfatória aos usuários. A base do desenvolvimento *web* é composta por JavaScript juntamente com HTML (*HyperText Markup Language*) e CSS (*Cascading Style Sheets*) (DUÒ, s.d.).

O JavaScript é uma linguagem de programação voltada para o desenvolvimento de sites e aplicativos *web* interativos. O JavaScript permite aos programadores a adição de recursos dinâmicos às páginas *web*, tais como animações, validação de formulários, dentre outros aspectos. É uma linguagem de alto nível que é interpretada diretamente no navegador do usuário (implementa funcionalidades em página *web* no lado cliente), o que significa que não precisa ser compilada antes de ser executada (WEB, s.d.).

HTML é a linguagem de marcação básica empregada na elaboração de páginas na *web*. Caracterizada como uma linguagem de natureza estática, o HTML confere definição à estrutura e ao conteúdo de uma página. Por meio do HTML, é viável a construção de parágrafos, listagens, cabeçalhos, inserção de imagens, links e demais componentes elementares indispensáveis para a apresentação de dados em um sítio eletrônico. Entretanto,

o HTML, em sua singularidade, não viabiliza a interatividade e o dinamismo exigidos por muitos *websites* contemporâneos (ALURA, s.d.).

Com a introdução do JavaScript, torna-se possível a manipulação de elementos HTML, a modificação de estilos, a exibição de alertas, a execução de operações aritméticas e a interação com o usuário. Em complemento, o CSS emerge como uma linguagem de estilo destinada a estabelecer a configuração visual de uma página *web*. Mediante o CSS, torna-se exequível a definição de tonalidades, tipos de fontes, esquemas de *layout* e outros atributos visuais de uma página. Com o CSS, é viável a distinção entre o conteúdo e a apresentação, contribuindo para a simplificação da manutenção e das modificações de um site (ALURA, s.d.).

3 METODOLOGIA

Este capítulo expõe a metodologia utilizada no desenvolvimento deste trabalho, descrevendo informações sobre a estratégia empregada para a regulagem e o desenvolvimento do instrumento didático.

Reitera-se que o objetivo deste trabalho é desenvolver a automação de ajustes de componentes da mesa tangível e a geração de um recurso educacional.

3.1 AJUSTES

3.1.1 Projeto Base

A mesa tangível projetada (Figura 2) para educação inclusiva é desenvolvida com uma especificação de baixo custo, variando entre R\$ 3.000 e R\$ 5.000, dependendo do projetor empregado (PREUSS, 2021). Na Tabela 1, está descrita a relação de componentes usados para montagem da mesa. Para o funcionamento desse dispositivo, utiliza-se um sistema de rastreamento baseado em visão computacional por meio das tecnologias reacTIVision e TUIO (KALTENBRUNNER, 2009).

O software reacTIVision representa uma solução gratuita que se encontra sob licença GPL (*General Public License*), permitindo o rastreamento de objetos tangíveis, empregando marcadores fiduciais e suporte a multi-toque. Essa ferramenta incorpora algoritmos de visão computacional (KALTENBRUNNER, 2009). Em conjunto com o protocolo TUIO, o reacTIVision disponibiliza um conjunto de recursos para a criação de Interfaces Tangíveis de Usuário (TUIs) com base em mesas tangíveis, fazendo uso de iluminação infravermelha difusa (DI) (PREUSS, 2021).

A mesa usa iluminação infravermelha (IR), especificamente empregando a técnica *Diffused Illumination* (SCHÖNING; HOOK; BARTINDALE et al., 2010), com dois iluminadores IR de 850 nm direcionados para a parte inferior da mesa. Isso resulta em uma iluminação indireta e dispersa sobre o tampo de acrílico, minimizando saturações e evitando reflexos indesejados. Essa configuração também impede que os iluminadores sejam captados diretamente pela câmera ou refletidos na superfície (PREUSS, 2021).

Os sistemas que empregam a técnica de DI (*Diffused Illumination*) usam um projetor e uma câmera sensível a infravermelho posicionada sob a superfície. No entanto, na abordagem DI, a fonte de iluminação infravermelha é posicionada sob a superfície de projeção, ocasionando uma iluminação clara com IR na área à frente da superfície (SCHÖNING; HOOK; BARTINDALE et al., 2010). Por meio desse método, a câmera identifica os reflexos de infravermelho provenientes de dedos e objetos presentes na superfície. A detecção de toque explora o fenômeno de que uma superfície de projeção (colocada sobre a superfície) dissemina a luz, resultando na difusão dos objetos distantes. A abordagem DI

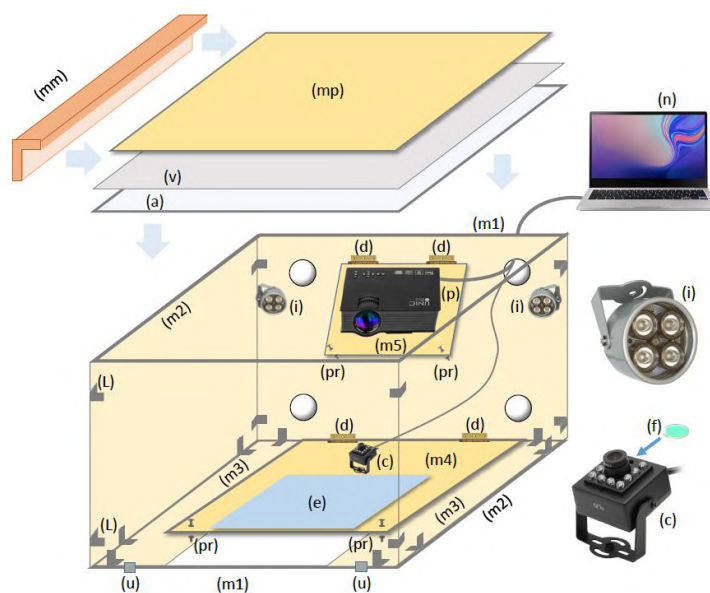
possui uma vantagem adicional ao facilitar o rastreamento de objetos físicos, que podem ser identificados por sua configuração ou através do uso de marcadores fiduciais. Tais marcadores são elementos facilmente discerníveis, geralmente demonstrando um padrão distinto em sua forma, fixados na base dos objetos.

Para a captação de imagens, uma *webcam* infravermelho HD USB, equipada com uma lente de 28 mm, suporte m12, é empregada. Um filtro passa-alta IR de 850 nm, com 12 mm, é posicionado entre a lente e o sensor da câmera. Tal arranjo garante que a câmera registre apenas os reflexos infravermelhos nos marcadores fiduciais dos objetos dispostos sobre a mesa. As câmeras infravermelho, que já incorporam LEDs (*Light Emitting Diodes*) infravermelhos, eliminam o filtro que bloqueia a entrada de IR no sensor, presente em *webcams* convencionais. Portanto, embora seja possível recorrer a uma *webcam* convencional, requer-se a remoção desse filtro e a aplicação do filtro passa-alta IR de 850 nm (PREUSS, 2021).

Em relação ao projetor, sua característica fundamental reside em ser de curta projeção e possuir a capacidade de projetar uma imagem com dimensões superiores a 36" (80 cm x 50 cm) a uma distância de 120 cm. Essa projeção, auxiliada por um espelho, preenche toda a largura da mesa. Adicionalmente, busca-se um projetor que tenha compatibilidade com o formato 16:9 (*widescreen*), a fim de otimizar a aplicação de sua luminosidade e o tamanho da projeção. A luminosidade mínima requerida é de 1.600 lumens para uso em um ambiente escurecido ou de 3.000 lumens para utilização em um ambiente iluminado. Existe a necessidade do projetor possuir uma rosca de fixação em sua base para fixação à estrutura da mesa. Um critério relevante ao selecionar um projetor é a existência de uma rosca de fixação em sua base, possibilitando sua fixação à estrutura da mesa. Devido ao fato de o projetor estar alojado dentro da mesa, a projeção pode criar sombras, limitando a área de projeção no tampo. Isso permite a utilização da parte traseira do tampo para dispor objetos tangíveis que não estejam em uso (PREUSS, 2021).

A confecção da mesa empregando MDF (*Medium Density Fiberboard*) confere resistência e facilidade de desmontagem e transporte. Uma moldura contornando o tampo serve para fixar o acrílico e evitar deslocamentos indesejados. Essa moldura orienta a aplicação de uma proteção sobre a superfície quando a mesa não está em uso. Na parte traseira da mesa, aberturas são inseridas para promover a ventilação e reduzir o acúmulo de calor gerado pelo projetor (PREUSS, 2021).

O tampo da mesa é constituído de acrílico, revestido com vinil translúcido, película para projeção em vitrines ou, como alternativa, papel vegetal coberto com vinil ou plástico PVC (*Polyvinyl chloride*) transparente (cristal), podendo ser simples ou contar com adesivos (*Contact*) em sua superfície. A opacidade do vinil deve ser suficiente para reter a projeção da imagem e, ao mesmo tempo, permitir que a câmera capture os marcadores fiduciais na base dos objetos sobre a mesa. Posicionar o vinil abaixo do acrílico distorcerá a imagem dos marcadores fiduciais e prejudicará o rastreamento dos objetos. Ao aplicar o vinil diretamente sobre o acrílico, o rastreamento é otimizado, embora seja necessário ajustar a posição dos iluminadores IR dentro da mesa para evitar que seus reflexos sejam captados pela câmera (PREUSS, 2021).

Figura 2: Mesa tangível utilizada.

Fonte: Preuss (2021)

Tabela 1: Componentes usados para montagem da mesa tangível.

Índex	Componente
a	Acrílico Transparente
c	Câmera IR
d	Dobradiças
e	Espelho
f	Filtro IR
i	Iluminador Infravermelho
L	Cantoneiras L
m1	MDF 15 mm (80 cm x 74 cm)
m2	MDF 15 mm (57 cm x 74 cm)
m3	MDF 15 mm (25 cm x 57 cm)
m4	MDF 15 mm (42 cm x 47 cm)
m5	MDF 15 mm (40 cm x 30 cm)
mm	Moldura de madeira 6 cm x 3 cm
mp	MDF 6 mm (78 cm x 58 cm)
n	Computador ou Notebook
p	Projektor
pm	Parafusos para Madeira 12 mm
pr	Parafusos 80 mm com Arruela e Porca
u	Sapata
v	Vinil translúcido

Fonte: Preuss (2021)

Esse projeto é empregado como referência para a montagem de uma nova estrutura com a automação da posição do espelho e da câmera infravermelho (Seção 3.1.3.1).

Adicionalmente, a nova mesa incorpora um sistema de ventilação (Seção 3.1.3.2) com o propósito de assegurar a manutenção da temperatura apropriada, evitando a formação de pontos quentes, conforme as especificações indicadas pelos fabricantes. O superaquecimento, se ocorrer, pode resultar na redução do desempenho e na diminuição da vida útil dos componentes. Um programa desenvolvido em Python (Seção 3.1.3.3) é executado no Raspberry Pi Zero W (Seção 3.1.2) para realizar essas automações.

3.1.2 Raspberry Pi Zero W

O Raspberry Pi é um minicomputador integrado em uma única placa, desenvolvido pela Raspberry Pi Foundation, equipado com arquitetura *system-on-a-chip* que integra a unidade central de processamento, unidade de processamento gráfico, memória e outros componentes essenciais. Foi concebido com a finalidade de ser compacto, acessível e versátil (MELLO, s.d.). Diferentes configurações de memória são fabricadas para proporcionar aos usuários flexibilidade na escolha, de acordo com suas necessidades.

O Raspberry Pi Zero W (Figura 18) é anunciado como uma versão compacta e acessível do computador de placa única Raspberry Pi, incorporando conectividade sem fio (Wi-Fi) e Bluetooth. Equipado com um processador de 1 GHz, 512 MB de RAM e portas micro-USB, possibilita a integração com diversos periféricos e sensores para atender a diversas aplicações (WIKIPÉDIA, s.d.). Adicionalmente, suporta vários sistemas operacionais, sendo o Raspbian, uma distribuição baseada em Linux especificamente projetada para o Raspberry Pi, o mais amplamente utilizado (MAGNETISM, s.d.).

Uma das aplicações mais proeminentes do Raspberry Pi Zero W está no campo da automação residencial. Com seus pinos GPIO (*General Purpose Input/Output*), pode se comunicar com sensores e atuadores (MAGNETISM, s.d.). Isso permite que usuários controlem e automatizem vários aspectos de suas casas, como iluminação, sistemas de segurança e até mesmo jardinagem. Além disso, com sua interface amigável e extensos recursos online, tornou-se uma ferramenta educacional valiosa para o ensino de programação, ciência da computação e eletrônica.

Adicionalmente, seu pequeno formato e opções de conectividade o tornam uma plataforma ideal para criar dispositivos IoT (Internet das Coisas) que podem coletar, processar e transmitir dados para a nuvem (WIKIPÉDIA, s.d.). De estações meteorológicas a sistemas de vigilância doméstica, as possibilidades são virtualmente infinitas.

3.1.3 Implementação

3.1.3.1 Posicionamento do Espelho e da Câmera Infravermelho

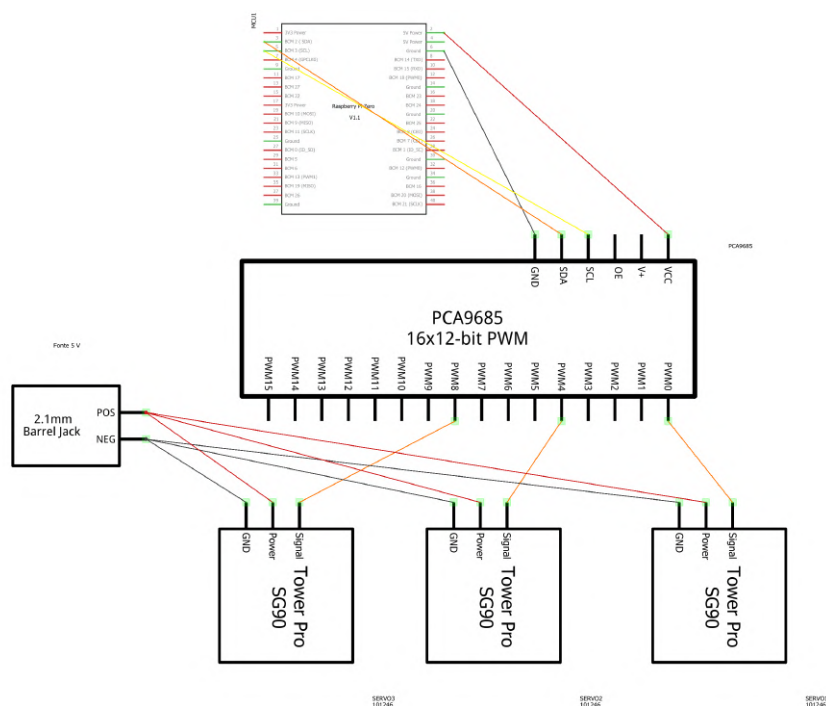
Com o propósito de automatizar a orientação do espelho, este é fixado em uma chapa, conectada a um eixo cujo ângulo é controlado pelo servo motor (no Apêndice A, Figura 19) em conjunto com o Raspberry Pi Zero W (Seção 3.1.2) e o módulo PWM PCA9685 (no Apêndice A, Figura 20). O servo motor representa um dispositivo amplamente empregado no âmbito da robótica e sistemas automatizados, visando o controle de movimentos precisos dentro de um ângulo específico. A técnica de ajuste angular utilizando um servo motor encontra aplicação frequente no campo da engenharia elétrica, sendo utilizada em variados contextos como robótica, modelagem tridimensional, gerenciamento de câmeras,

manipuladores robóticos, entre outros. O espelho é fixado em uma chapa ligada a um eixo controlado pelo servo motor.

A câmera infravermelho é instalada em um suporte *Pan Tilt* (no Apêndice A, Figura 21) para controle da sua posição. O dispositivo denominado suporte *Pan Tilt* é amplamente empregado no âmbito da visão computacional. Ele consiste em um mecanismo de controle de movimento que viabiliza tanto a rotação horizontal (*pan*) quanto a vertical (*tilt*) de câmeras e outros dispositivos utilizados em diversas aplicações. Composto por dois eixos de rotação, um destinado ao movimento horizontal e outro ao movimento vertical, esse suporte é governado por motores que conferem precisão ao posicionamento do dispositivo acoplado. Além disso, é possível equipá-lo com sensores, como *encoders*, que fornecem *feedback* ao sistema de controle, permitindo a monitoração e ajuste contínuo da posição. No cenário industrial, o suporte *Pan Tilt* é utilizado para o controle de máquinas e equipamentos automatizados. Essa tecnologia permite o direcionamento do sensor de um robô, por exemplo, para identificar corretamente peças ou pontos de interesse em uma linha de produção (IANNI, 2023). A versatilidade desse dispositivo possibilita sua integração a outros sistemas, tais como sensores de proximidade ou câmeras infravermelhas, ampliando assim sua aplicabilidade e utilidade. Por meio desse suporte, é factível realizar varreduras horizontais e verticais, bem como rastrear objetos ou indivíduos em movimento.

A manipulação dos servos motores ocorre mediante a geração de pulsos PWM (*Pulse Width Modulation*), que consistem em sinais elétricos com variações na largura dos pulsos, exercendo influência sobre a posição do eixo motorizado. Conectam-se os cabos de sinal, de alimentação e de aterramento dos servos motores nas saídas do módulo PWM PCA9685. Adicionalmente, uma fonte externa de 5 V é conectada ao módulo para servir como alimentação para os servos motores. Esse módulo é diretamente conectado à Raspberry Pi Zero W conforme Figura 3 e Figura 25 (Apêndice B).

Figura 3: Esquemático elétrico para automação da posição do espelho e da câmera infravermelho.



Fonte: do autor.

O módulo PWM PCA9685 é um dispositivo que disponibiliza vários canais de saída PWM (Pulse Width Modulation) por meio da interface de comunicação I2C (Inter-Integrated Circuit). Este módulo é frequentemente empregado em projetos eletrônicos para gerenciar a intensidade de luz, velocidade de motores, ou outras aplicações que demandam controle de potência por meio de sinais PWM. O PCA9685 comumente oferece 16 canais PWM independentes, possibilitando o controle de até 16 dispositivos ou circuitos distintos (AGUIAR et al., 2023). A faixa típica de frequência de atualização varia de 40 Hz a 1.000 Hz, e a frequência do sinal PWM pode ser ajustada conforme os requisitos específicos da aplicação.

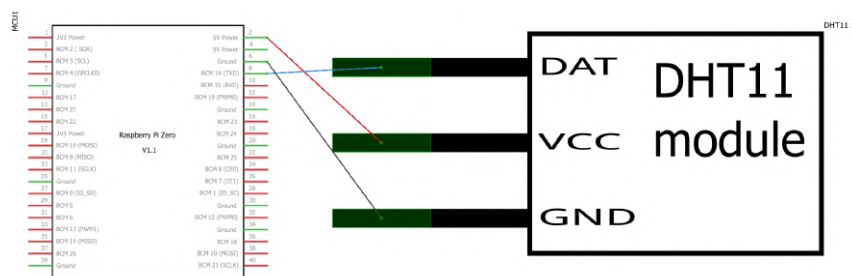
3.1.3.2 Sistema de Ventilação

A fim de implementar o sistema de ventilação, o módulo DHT11 (no Apêndice A, Figura 22) é usado. O módulo DHT11 é um sensor de temperatura e umidade amplamente empregado em projetos de eletrônica e automação residencial. Com um design compacto, esse dispositivo é facilmente integrado em diversos sistemas. O sensor DHT11 é composto por um sensor de umidade capacitivo e um sensor de temperatura do tipo termistor NTC, ou seja, um resistor sensível a variações de temperatura. A medida de umidade ocorre dentro da faixa de 20% a 80%. No que diz respeito à temperatura, as medidas ocorrem entre 0 °C a 50 °C. A alimentação pode ocorrer entre 3,5 V a 5 V, com uma corrente máxima de 2,5 mA. Dentro do módulo DHT11, um microcontrolador captura as medições e transmite os valores no formato digital por meio de um pino de saída (FREITAS MARIANO, s.d.).

Com o intuito de garantir um fluxo de ar consistente, contribuindo para a dissipação do calor gerado pelos componentes internos da estrutura, são utilizados dois *coolers* (no Apêndice A, Figura 23) com dimensões de 12 cm x 12 cm x 3,8 cm e alimentação de 12 V. A alimentação dos *coolers* é gerenciada pelo módulo relé de 8 canais (no Apêndice A, Figura 24). Esse módulo tem a capacidade de controlar até 8 dispositivos, podendo ser tanto de corrente contínua quanto de corrente alternada, desde que estejam dentro do limite de corrente de 10 A. O relé opera mediante o comando de um microcontrolador, recebendo uma tensão de 5 V e realizando o chaveamento para uma das posições que ele opera, Normal Aberto, Normal Fechado e Comum, para acionar cargas. O módulo relé apresenta a vantagem de possuir um optoacoplador, componente cuja função é isolar uma região da outra para proteger o microcontrolador em caso de sobrecarga de energia (CANALS, s.d.).

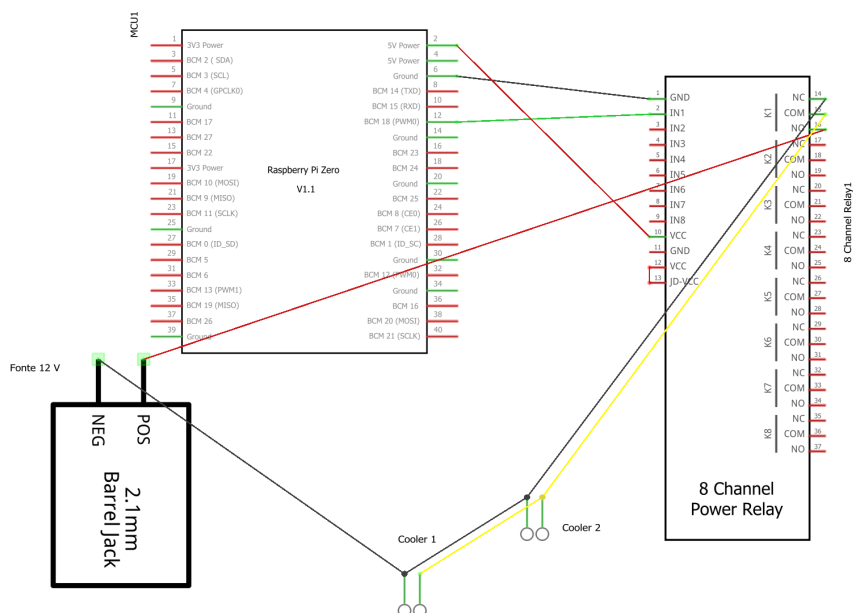
Portanto, para o sistema de ventilação, os cabos de sinal, de alimentação e de aterramento do módulo DHT11 são diretamente conectados à Raspberry Pi Zero W conforme Figura 4 e Figura 26 (Apêndice B). Além disso, uma porta da Raspberry Pi Zero W comanda o relé para habilitar a alimentação de uma fonte externa de 12 V, ativando os *coolers*, conforme Figura 5 e Figura 27 (Apêndice B).

Figura 4: Esquemático elétrico do uso do módulo DTH11 e do Raspberry Pi Zero W para automação do sistema de ventilação.



Fonte: do autor.

Figura 5: Esquemático elétrico do uso dos coolers, do módulo relé de 8 canais, da fonte externa de 12 V e do Raspberry Pi Zero W para automação do sistema de ventilação.



Fonte: do autor.

3.1.3.3 Desenvolvimento em Python

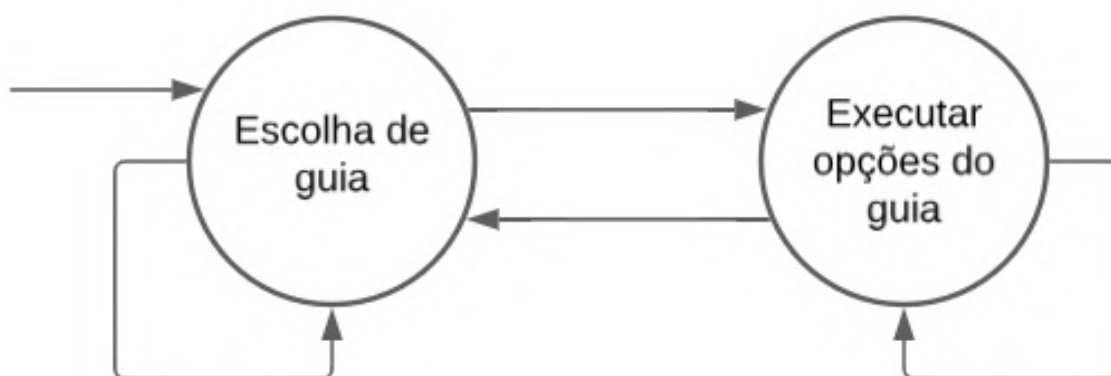
A elaboração do programa para controlar os circuitos montados e a criação de uma interface gráfica são viabilizadas por meio da utilização de várias bibliotecas em Python (Apêndice C). PyQt5 e PySide2 oferecem uma diversidade de ferramentas e recursos que facilitam a criação de aplicativos com interfaces intuitivas e visualmente atraentes, incluindo o desenvolvimento de *layouts* e animações. Para possibilitar a comunicação adequada entre o Raspberry Pi Zero W e os módulos DHT11 e PWM PCA9685, bibliotecas específicas (Adafruit) são instaladas e incorporadas no código.

Além disso, para o correto funcionamento do módulo PWM PCA9685, é necessário ativar a conexão I2C, um protocolo de comunicação serial bidirecional que permite a troca de dados entre dispositivos eletrônicos utilizando apenas dois fios: um para a transmissão de

dados (SDA - *Serial Data*) e outro para o sinal de *clock* (SCL - *Serial Clock*). Este protocolo é empregado para conectar vários dispositivos em um barramento comum, permitindo a comunicação entre eles. Cada dispositivo I2C possui um endereço único, simplificando a identificação e a comunicação com um dispositivo específico na rede.

O programa é desenvolvido (Figura 6) para assegurar que a troca do guia escolhido (guias: “Menu”, “Sistema de Ventilação”, “Ajustar Infravermelho”, “Ajustar Espelho”, “Ajuda” e “Sair”) esteja sempre disponível. Dessa forma, possibilita ao usuário escolher entre executar a opção oferecida ou alterar o guia.

Figura 6: Máquina de estados para demonstrar o funcionamento do programa (“Escolha de guia” como estado inicial).



Fonte: do autor.

3.2 RECURSO EDUCACIONAL

3.2.1 Eduba Editor e Nidaba Player

O Eduba Editor é uma ferramenta que viabiliza a criação de material educacional envolvendo cenas, imagens, textos, animações e atividades voltadas à mesa tangível. Essa criação ocorre de maneira intuitiva e interativa, tanto em computadores quanto em dispositivos móveis. No ambiente do editor, as aplicações são construídas através de um agrupamento de cenas. Em cada cena, são selecionados elementos como imagem de fundo e áudio de narração ou explicação. Pode-se ainda incorporar outros componentes, a exemplo de texto, imagens com a capacidade de animação, áreas de interação baseadas nos marcadores fiduciais, vídeos e mecanismos de *feedback*. Quando se estabelece a interação com objetos tangíveis, as regiões de interação são delimitadas dentro do editor. Além disso, são configurados os elementos fiduciais adequados e inadequados, bem como o respectivo retorno sonoro ou pictográfico para cada resposta (PREUSS, 2021).

A estruturação do Eduba Editor foi realizada por meio das linguagens HTML5, CSS, JavaScript e PHP (*Hypertext Preprocessor*), visando à operacionalização no navegador (Chrome, Edge, Firefox e Safari). Ele estabelece comunicação com o software reacTIVision por meio do protocolo TUIO sobre *websockets* (reconhecimento das interações dos marcadores fiduciais posicionados na mesa). As aplicações são salvas no formato JSON, acompanhadas

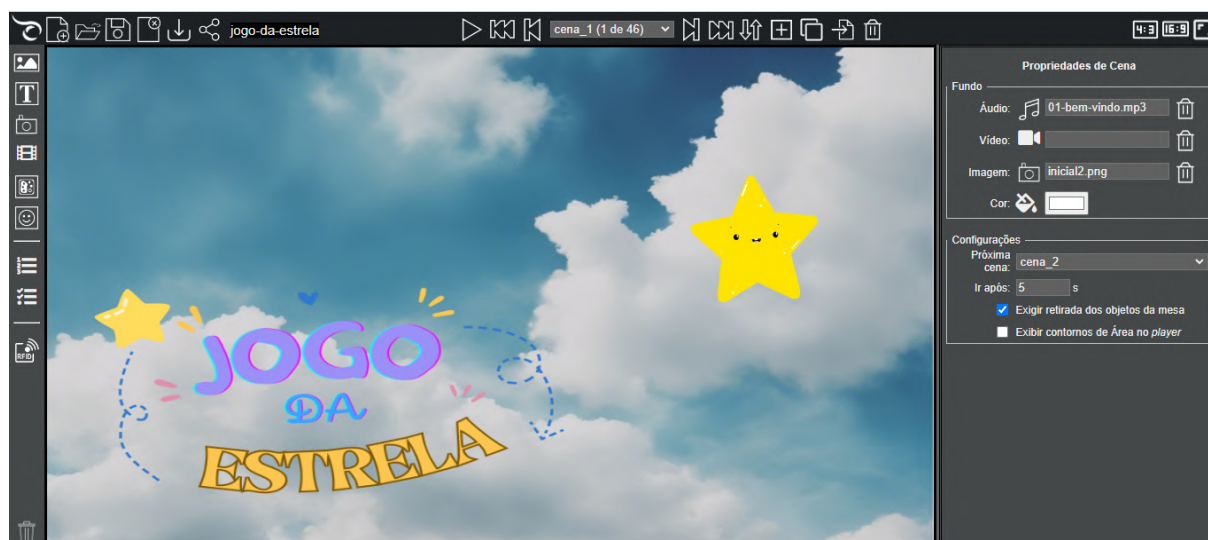
dos arquivos de mídia (orientação a objetos). Isso permite que uma aplicação em formato HTML5 seja criada, destinada à aplicação na mesa tangível ou no simulador em dispositivos móveis e computadores. O editor dispõe de uma interface baseada no conceito *What You See Is What You Get* (O que você vê é o que você obtém). Esse ambiente permite a programação das atividades sem a necessidade de codificação explícita. Os elementos são colocados de forma interativa por meio do uso do cursor: ação de clique, arraste e redimensionamento (PREUSS, 2021).

O Nidaba Player é o módulo da plataforma que desempenha a função de executar as aplicações criadas no Eduba Editor em uma mesa tangível. Esse módulo realiza a projeção das cenas na superfície da mesa e reage às colocações e movimentações dos mecanismos. Isso acontece através da comunicação com o reacTIVision (software de visão computacional responsável pelo rastreamento dos marcadores fiduciais fixados na base dos objetos). O Nidaba Player controla as interações do usuário e os retornos sonoros e visuais correspondentes a cada objeto posicionado na mesa. Além disso, avalia o momento em que todos os fiduciais foram dispostos nas áreas correspondentes, indicando a exibição da próxima cena ou de uma cena específica (PREUSS, 2021).

3.2.2 Elaboração

Para o desenvolvimento do jogo, inicialmente, uma clínica de fonoaudiologia focada no tratamento de crianças com TEA é visitada. O conhecimento teórico acerca dos sintomas e das características peculiares do Transtorno do Espectro Autista auxilia na elaboração da proposta e das funcionalidades do jogo. As diferentes abordagens de tratamento são fontes para a criação do Jogo da Estrela, cuja tela inicial no Eduba Editor está exibida na Figura 7.

Figura 7: Edição da tela inicial no Eduba Editor.



Fonte: do autor.

Na edição do jogo (Figura 8), utiliza-se imagens isentas de direitos autorais para compor a cena. Além disso, são delimitadas áreas de interação (mediante o reconhecimento de fiduciais presentes nos cartões físicos) com o usuário e de *feedback*. Áudios são incorporados para proporcionar uma eficiente comunicação.

Figura 8: Demarcação de área de interação com o usuário em vermelho e área de feedback em azul.



Fonte: do autor.

4 RESULTADOS

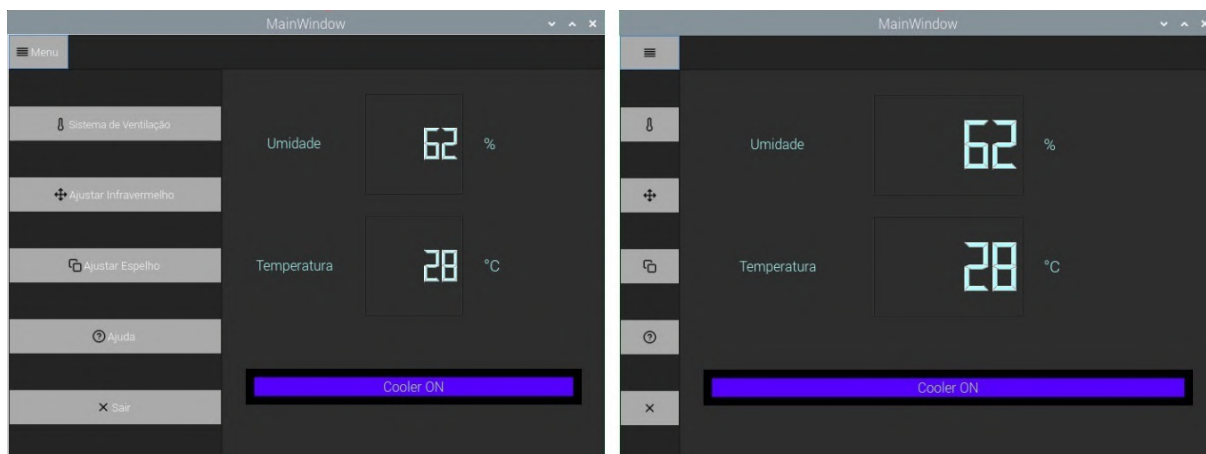
A presente seção aborda os resultados obtidos. A automação desenvolvida engloba o ajuste da posição da câmera infravermelho e do espelho. Além disso, abrange o controle do sistema de ventilação implementado. No recurso didático desenvolvido, são utilizados três diferentes enfoques para interação.

4.0.1 Programa e Componentes

A interface gráfica é desenvolvida para possibilitar a interação do usuário com um programa por meio de elementos visuais, como botões, ícones, menus e janelas, em lugar de comandos de texto. Essa abordagem simplifica a comunicação entre o usuário e o software, proporcionando uma experiência mais intuitiva e amigável. Dessa forma, a interface capacita o usuário a comandar, de forma facilitada, tarefas que usam algoritmos avançados.

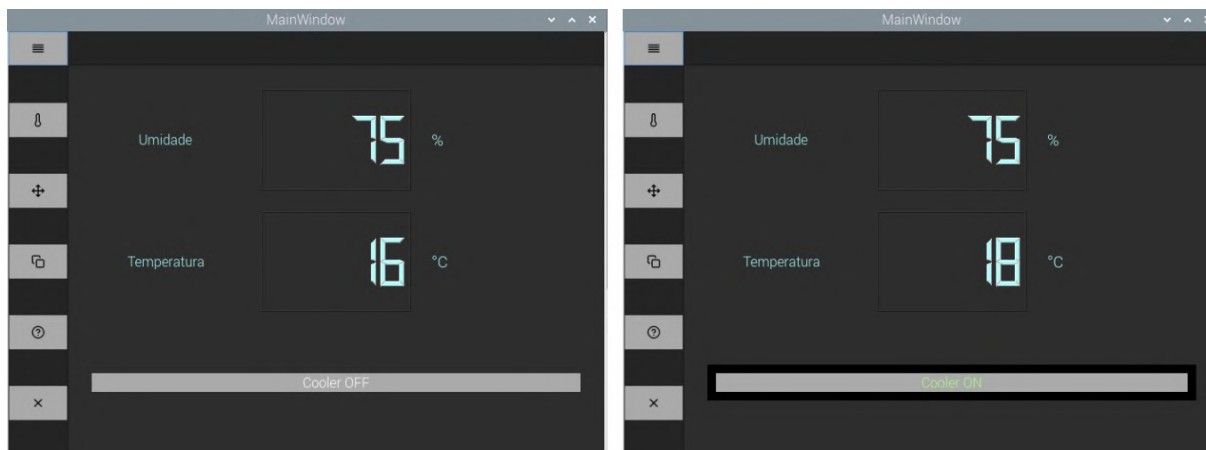
A interface é implementada em Python e oferece diversas funcionalidades. Em todos os guias, existe um símbolo para facilitar a identificação. Ao acionar o guia “Menu”, ocorre uma alternância entre os dois *layouts* disponíveis conforme a Figura 9. No guia “Sistema de Ventilação”, é possível verificar a umidade e temperatura detectadas pelo módulo DTH11. Para valores iguais ou inferiores a 20 °C (Figura 10), é concedida ao usuário a opção de ligar ou desligar o sistema de ventilação. Para valores superiores a essa temperatura, o sistema de ventilação é ativado automaticamente. A fim de simplificar essa compreensão, todos os botões disponíveis para o usuário interagir compartilham características semelhantes, como a “cor de fundo”. Quando o botão não está acessível ao usuário, essa característica é modificada (como o botão “Cooler ON” na Figura 9).

Figura 9: *Layouts disponíveis: com o nome dos guias (imagem da esquerda) e sem o nome dos guias (imagem da direita).*



Fonte: do autor.

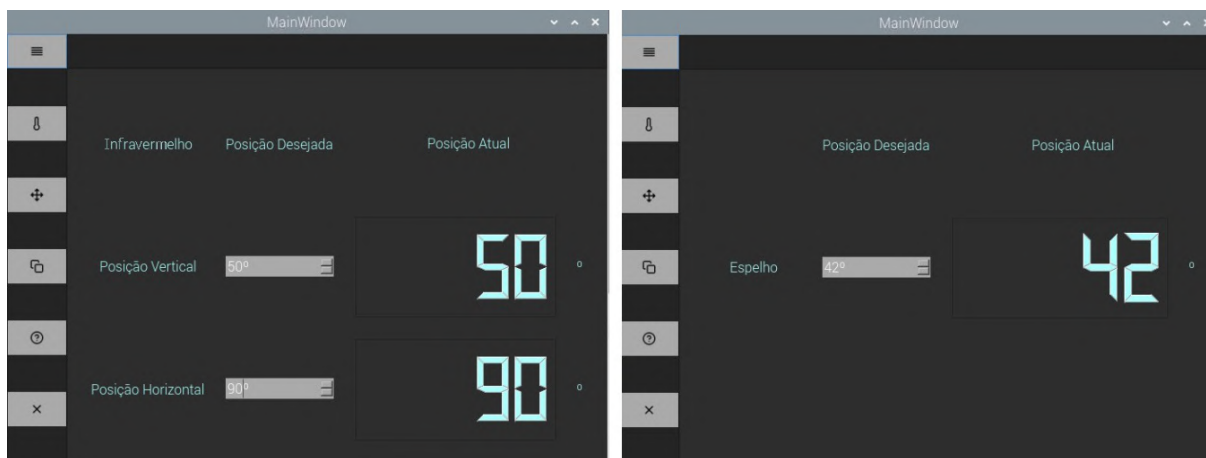
Figura 10: *Sistema de ventilação para temperaturas inferiores a 20 °C (sistema desligado na imagem da esquerda e sistema ligado na imagem da direita).*



Fonte: do autor.

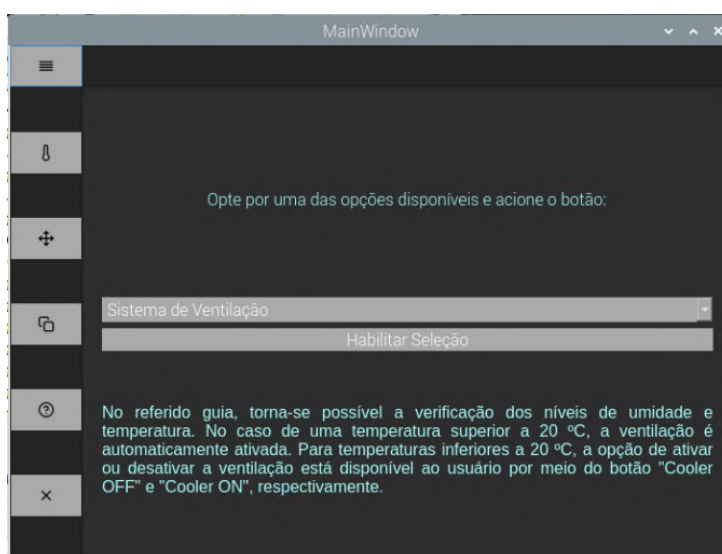
Nos guias “Ajustar Infravermelho” e “Ajustar Espelho”, são disponibilizados campos para inserção da posição desejada e visualização da posição atual dos servos motores (Figura 11). Adicionalmente, no guia “Ajuda”, é possível acessar informações sobre o funcionamento do programa (Figura 12) e o guia “Sair” destina-se ao encerramento do programa.

Figura 11: Guias “Ajustar Infravermelho” (imagem da esquerda) e “Ajustar Espelho” (imagem da direita).



Fonte: do autor.

Figura 12: Guia “Ajuda”.



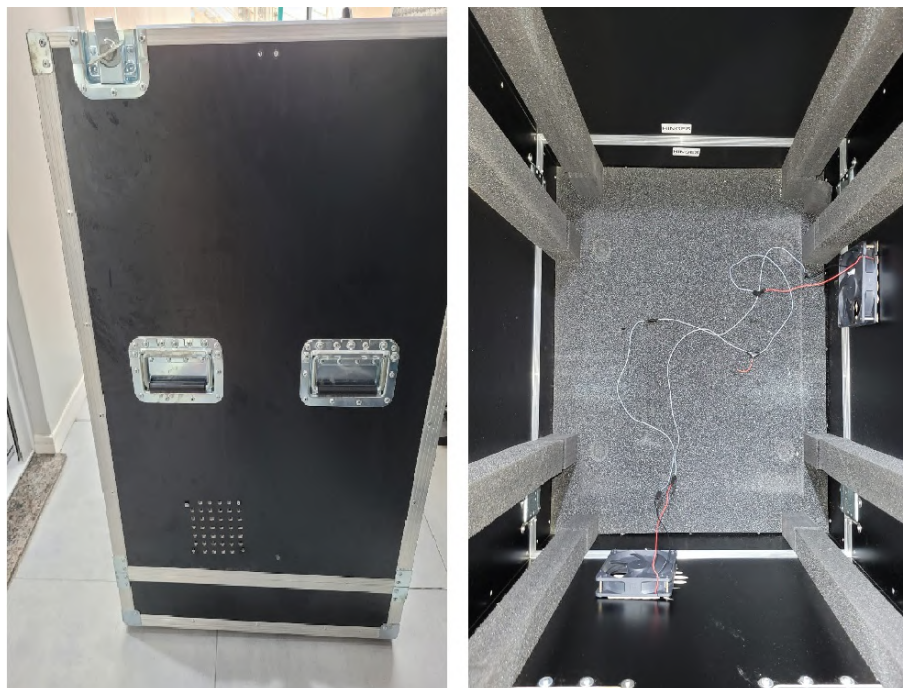
Fonte: do autor.

O sistema de ventilação encontra-se instalado na parte inferior, na região destinada ao controlador, da caixa (Figura 13). O projetor não está disponível para a montagem da estrutura. Dessa forma, a montagem da mesa não pode ser realizada, uma vez que é essencial testar a disposição ideal de todos os componentes para garantir o melhor aproveitamento da projeção. A câmera infravermelho, montada com o suporte *Pan Tilt*, é apresentada na Figura 14.

Adicionalmente, o ajuste da posição do espelho (Figura 15) é realizado por meio da modificação do ângulo do eixo (64 cm de comprimento e 1,5875 cm de espessura) ligado a um rolamento rígido de esfera (2 cm x 4,2 cm x 1,2 cm - Modelo: 6004 2RS). A parte externa desse rolamento está soldada em uma placa (7 cm x 7 cm x 0,1 cm) para fixação na caixa. No outro lado, outro rolamento é empregado como ponto de apoio do eixo. Uma solda é aplicada para realizar a conexão entre esse rolamento e a estrutura (parte conectada

ao rolamento: 4 cm x 3 cm x 0,2 cm - parte em paralelo ao eixo: 15 cm x 3 cm x 0,2 cm) localizada abaixo do servo motor. Esse suporte está soldado em outra placa (5 cm x 3 cm x 0,2 cm) para fixação na caixa e possui dois guias (8 cm x 2 cm x 0,2 cm) para manter a orientação do servo motor. A variação da posição é efetuada pelo servo motor montado com a estrutura utilizada como base no suporte *Pan Tilt* e conectado ao eixo por meio de uma placa (4,5 cm x 4 cm x 0,2 cm) soldada.

Figura 13: Sistema de ventilação: visão externa (imagem da esquerda) e visão interna (imagem da direita).



Fonte: do autor.

Figura 14: Câmera infravermelho com o suporte *Pan Tilt*.



Fonte: do autor.

Figura 15: Na imagem superior, a estrutura encontra-se montada com o servo motor, conectada em dois pedaços de madeira para simular a caixa. Na imagem inferior, a estrutura está montada com o servo motor e o espelho.



Fonte: do autor.

4.0.2 Jogo da Estrela

O jogo contém três abordagens: pareamento, cor e fonema (No Apêndice D, Figura 29). Na trilha de pareamento (No Apêndice D, Figura 30), existem duas categorias (animais e esportes), e o objetivo é que a identificação do cartão contendo a figura igual à exibida na tela. Este jogo estimula o reconhecimento visual, auxiliando a criança a associar imagens semelhantes e aprimorar suas habilidades de discriminação visual (No Apêndice D, figuras 31 a 40).

Na trilha de cor (Figura 41), existem duas categorias (animais e objetos), e a criança deve posicionar o cartão com a cor da figura divulgada. Associar imagens a cores específicas implica na compreensão e aplicação de conceitos de cores, contribuindo para o desenvolvimento da percepção cromática e associação visual (No Apêndice D, figuras 42 a 51).

Na trilha de fonema (Figura 52), é proposto que a criança ouça o áudio com a palavra relacionada à imagem e faça a escolha da grafia correspondente. A alfabetização de um indivíduo não se constrói exclusivamente por meio de apenas um método ou ferramenta, mas sim por meio de diferentes conjuntos de atividades, métodos, ambientes e ferramentas. A capacidade de refletir sobre os sons da fala e identificar seus correspondentes gráficos é necessária no desenvolvimento da leitura e da escrita (BIGOCHINSKI; ECKSTEIN, 2016). O jogo (No Apêndice D, figuras 53 a 72) propõe trabalhar de maneira significativa

a consciência fonológica no processo de alfabetização, a aquisição e a apropriação da linguagem escrita, diferenciando significados e significantes e compreendendo a relação entre a linguagem oral e escrita. O jogo busca ser uma ferramenta para que as fonoaudiólogas possam demonstrar às crianças a dimensão sonora das palavras. A consciência fonológica é a habilidade de entender e manipular os sons da linguagem falada. Envolve a capacidade de identificar, segmentar e manipular os sons individuais (fonemas) dentro das palavras. Isso inclui a capacidade de reconhecer rimas, segmentar palavras em sílabas, identificar sons iniciais e finais de palavras, entre outras habilidades relacionadas ao processamento auditivo da linguagem (ALVES, 2009).

O recurso educacional é gerado com cores e ilustrações agradáveis, evitando estímulos concorrentes e buscando uma estrutura clara e objetiva. As telas também mantêm consistência entre si, de modo que não seja necessário compreender vários padrões e formas de interações diferentes. Além disso, são incluídas imagens e áudios para *feedback* de erro e acerto (Figura 16).

Figura 16: Simulações no Eduba Editor.



Fonte: **do autor.**

Participar desse jogo exige que a criança mantenha a atenção na tarefa, contribuindo potencialmente para o desenvolvimento de habilidades de concentração e foco. Ao manipular os cartões (Figura 17) e posicioná-los na tela, a criança está aprimorando habilidades motoras finas, tais como destreza manual e coordenação olho-mão. Em todas as trilhas, após cinco perguntas acertadas, o jogo indica que a criança obteve êxito (No Apêndice D, Figura 73). Em todas as trilhas, as crianças têm a oportunidade de interagir com a fonoaudióloga. Essas ocasiões proporcionam à criança expressar verbalmente suas escolhas, fazer comentários sobre as imagens, cores ou palavras, fomentando o desenvolvimento da linguagem. Adicionalmente, à medida que a criança adquire proficiência nas regras do jogo e compreende como jogar de maneira independente, ela desenvolve um senso de autonomia e confiança.

Figura 17: Os cartões destinados ao jogo apresentam imagens de um lado para interação (imagem da esquerda) e fiduciais no outro (imagem da direita).



Fonte: do autor.

5 CONCLUSÕES

O presente estudo abordou a automatização dos ajustes dos componentes de uma mesa tangível e da elaboração de um jogo destinado a ser empregado como recurso educacional. A referida mesa é utilizada no contexto do tratamento de crianças que apresentam o Transtorno do Espectro Autista.

As políticas públicas de educação asseguram o direito ao acesso à educação para todas as pessoas, independentemente de possuírem ou não deficiência. Nesse contexto, a responsabilidade de adaptar, preparar e desenvolver técnicas, metodologias e recursos que garantam uma educação completa para esses estudantes recai sobre as instituições de ensino e os educadores.

A utilização de mesas tangíveis para fins educativos apresenta vantagens significativas, visto que permite interações com ambientes computacionais por meio da manipulação de objetos reais. Isso viabiliza atividades mais enriquecedoras e interativas, contribuindo para aumentar o engajamento e a motivação dos alunos. O aperfeiçoamento do sistema da mesa tangível se faz necessária para torná-la mais adequada ao contexto da educação personalizada e da gestão pedagógica.

Dentre os recursos apresentados, destaca-se o Raspberry Pi Zero W como uma plataforma de prototipagem eletrônica versátil. Sua arquitetura acessível e a extensa comunidade de usuários têm impulsionado sua relevância tanto na área educacional quanto na indústria. Com esse minicomputador integrado é possível realizar um interface de interação para o usuário controlar os ajustes apontados neste trabalho.

Devido à falta de um projetor, não foi possível realizar a montagem da mesa tangível. Apesar disso, foi constatado que as estruturas e os componentes eletrônicos utilizados para a regulação da posição do espelho, para o ajuste da câmera infravermelho e para o sistema de ventilação demonstraram eficiência. O programa desenvolvido proporcionou ao usuário o controle da automação.

O Eduba Editor demonstrou eficácia na elaboração de instrumentos educativos voltados para o tratamento de crianças autistas. Adicionalmente, diversas atividades podem ser desenvolvidas nessa ferramenta para outros públicos-alvo. Com três abordagens distintas, o jogo criado visou estimular o reconhecimento visual e a associação a diferentes imagens e cores, além de desenvolver a consciência fonológica.

REFERÊNCIAS

- AGUIAR, G. D. O. et al. Projeto e desenvolvimento de um simulador de irradiância solar para cubesats: uma alternativa de baixo custo. Florianópolis, SC., 2023.
- ALURA. *Desenvolvimento Web Com Html, CSS E Javascript*. [S.l.: s.n.]. Disponível em: <<https://www.alura.com.br/apostila-html-css-javascript>>. Acesso em: 4 jul. 2023, 20:57.
- ALVES, U. K. O Que É Consciência Fonológica. *Lamprecht R, Dutra APB, Scherer APR, Barreto FM, Alves UK, Santos RMS et al. Consciência dos sons da língua: subsídios teóricos e práticos para alfabetizadores, fonoaudiólogos e professores de língua inglesa. Porto Alegre: EDIPUCRS*, p. 31–46, 2009.
- ASSOCIATION, A. P. *Diagnostic and Statistical Manual of Mental Disorders*. 5th. Arlington, VA: American Psychiatric Publishing, 2013.
- AWARI. *o que É Python: a Linguagem de Programação Essencial para o Mercado de Tecnologia*. [S.l.: s.n.]. Disponível em: <https://awari.com.br/o-que-e-python-a-linguagem-de-programacao-essencial-para-o-mercado-de-tecnologia-3/?utm_source=blog&utm_campaign=projeto+blog&utm_medium=o%20que%20%C3%89%20Python:%20a%20Linguagem%20de%20Programa%C3%A7%C3%A3o%20Essencial%20para%20o%20Mercado%20de%20Tecnologia>. Acesso em: 18 out. 2023, 20:10.
- BARON-COHEN, S.; LOMBARDO, M. V. Autism and talent: the cognitive and neural basis of systemizing. *Dialogues in Clinical Neuroscience*, v. 19, n. 4, p. 345–353, dez. 2017. DOI: 10.31887/DCNS.2017.19.4/sbaroncohen.
- BAUMINGER, N.; KASARI, C. Loneliness and Friendship in High-Functioning Children with Autism. *Child Development*, v. 71, n. 2, 2000.
- BIGOCHINSKI, E.; ECKSTEIN, M. P. W. A importância do trabalho com a consciência fonológica para a aprendizagem da leitura e da escrita. *Revista Eletrônica do Curso de Pedagogia das Faculdades OPET*, p. 44–67, 2016.
- BORGES, L. E. *Python para Desenvolvedores - 2ª Edição*. [S.l.]: Edição do Autor, 2010. PDF.
- CANAIS, M. R. -. 8. *Mini Ventilador 120x120x38 - 12V*. [S.l.: s.n.]. Disponível em: <<https://curtocircuito.com.br/modulo-rele-5v-10a-8-canaiss.html>>. Acesso em: 17 out. 2023, 21:15.
- CARBAJAL, M. L. *Design e desenvolvimento de um ambiente de programação tangível de baixo custo para crianças*. 2016. Diss. (Mestrado) – Repositório UNICAMP, Campinas.

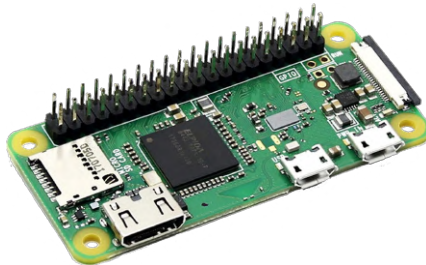
- COSTANZA, E.; ROBINSON, J. A region adjacency tree approach to the detection and design of fiducials. *Geneve: vision, video and graphics*, 2003. Accessed on 25 June 2017. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.116.1247&rep=rep1&type=pdf>>.
- DUÒ, M. *O Guia para Iniciantes em Design Responsivo da Web (Amostras de Código e Exemplos de Layout)*. [S.l.: s.n.]. Disponível em: <<https://kinsta.com/pt/blog/design-responsivo-web/>>. Acesso em: 4 jul. 2023, 20:55.
- FOUNDATION, P. S. *The Python Profilers*. [S.l.: s.n.]. Disponível em: <<https://docs.python.org/3/library/profile.html>>. Acesso em: 18 out. 2023, 20:40.
- FREEMAN, C.; LOUCA, F. *As time goes by: From the industrial revolutions to the information revolution*. [S.l.]: Oxford University Press, 2001.
- FREITAS MARIANO, E. DE. Aplicação didática de protótipos construídos em Plataforma Arduino.
- GLUZ, J. et al. Ambiente virtual tangível para integração sensorial no ensino de ciências numa perspectiva inclusiva. In: SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. ANAIS do Simpósio Brasileiro de Informática na Educação–SBIE. Fortaleza: [s.n.], 2018. P. 545–554.
- GRAY, C. A. *The New Social Story Book: Over 150 Social Stories that Teach Everyday Social Skills to Children with Autism or Asperger’s Syndrome*. [S.l.]: Future Horizons, 2010.
- HIBRIDA, E. *Avanços na Robótica: Da Inteligência Artificial à Automação Avançada*. [S.l.: s.n.]. Disponível em: <<https://materialpublic.imd.ufrn.br/curso/disciplina/1/63/1/2>>. Acesso em: 2 jun. 2023, 19:02.
- HOBBSAWM, E. J. *A Era das Revoluções: Europa 1789-1848*. [S.l.]: Paz e Terra, 2018.
- HODGETTS, S.; ZWAIGENBAUM, L.; NICHOLAS, D. Profile and predictors of service needs for families of children with autism spectrum disorders. *Autism Research*, v. 10, n. 12, p. 2006–2017, 2017.
- HOWLIN, P. Outcome in High-Functioning Adults with Autism with and without Early Language Delays: Implications for the Differentiation between Autism and Asperger Syndrome. *Journal of Autism and Developmental Disorders*, v. 33, n. 1, 2003.
- IANNI, D. Integrando Visão Computacional e Sistemas Embarcados: Rastreamento de objetos com CSRT e controle automático de câmera Pan-Tilt usando ESP32. Guaratinguetá, São Paulo - Brasil, 2023.
- IMD. *Aula 01 - Introdução aos problemas de controle*. [S.l.: s.n.]. Disponível em: <<https://materialpublic.imd.ufrn.br/curso/disciplina/1/63/1/2>>. Acesso em: 2 jun. 2023, 19:32.
- JANEKOVÁ, J.; FABIANOVÁ, J.; KÁDÁROVÁ, J. Optimization of the Automated Production Process Using Software Simulation Tools. *Processes*, v. 11, n. 2, 2023. ISSN 2227-9717. DOI: 10.3390/pr11020509. Disponível em: <<https://www.mdpi.com/2227-9717/11/2/509>>.
- JOHNSON, C. P.; MYERS, S. M. Identification and Evaluation of Children with Autism Spectrum Disorders. *Pediatrics*, v. 120, n. 5, p. 1183–1215, 2007.

- KALTENBRUNNER, M. reacTIVision and TUIO: A Tangible Tabletop Toolkit. In: ANAIS da Conferência. New York, NY, USA: ACM, 2009. P. 9–16. DOI: 10.1145/1731903.1731906. Disponível em: <<http://doi.acm.org/10.1145/1731903.1731906>>.
- LEBRUN, Y. et al. Simulation on an RFID Interactive Tabletop with Tangible Objects of Future Working Conditions: Prospects for Implementation in the Hospital Sector. In: [s.l.: s.n.], mar. 2022. P. 115–127. ISBN 9781786307996. DOI: 10.1002/9781119902614.ch8.
- MAGNETISM, E. .-. *O que é um Raspberry Pi?* [S.l.: s.n.]. Disponível em: <<https://www.electricity-magnetism.org/pt-br/o-que-e-um-raspberry-pi/>>. Acesso em: 17 out. 2023, 20:45.
- MELLO, M. *O que é Raspberry Pi, para que serve e principais modelos no Brasil.* [S.l.: s.n.]. Disponível em: <[https://victorvision.com.br/blog/o-que-e-raspberry-pi/#:~:text=Raspberry%20Pi%20\(RPi\)%20%C3%A9%20um,mouse%2C%20teclado%20e%20outros%20dispositivos](https://victorvision.com.br/blog/o-que-e-raspberry-pi/#:~:text=Raspberry%20Pi%20(RPi)%20%C3%A9%20um,mouse%2C%20teclado%20e%20outros%20dispositivos)>. Acesso em: 17 out. 2023, 20:40.
- OGATA, K. *Engenharia de Controle Moderno.* [S.l.]: LTC Editora, 2016.
- PASSERINO, L.; SANTAROSA, L. M. C. A Interação Social em Ambientes Telemáticos. *Informática na Educação: Teoria e Prática*, v. 5, n. 2, nov. 2002.
- PREUSS, E. *NIDABA: Plataforma Digital para Produção de Recursos Educacionais Inclusivos Baseados em Mesa Tangível.* 2021. Tese de Doutorado – Centro de Estudos Interdisciplinares em Novas Tecnologias na Educação, Programa de Pós-Graduação em Informática na Educação - Universidade Federal do Rio Grande do Sul, Porto Alegre. Disponível em: <<http://hdl.handle.net/10183/229757>>.
- PSIQUIATRIA, A. P. E. *Arduino na educação: uma nova geração de makers.* [S.l.: s.n.]. Disponível em: <<https://acaciapsi.com.br/transtorno-espectro-autista/#:~:text=0%20transtorno%20do%20espectro%20autista%20tamb%C3%A9m%20%C3%A9%20definido%20por%20padr%C3%B5es,capacidade%2C%20interven%C3%A7%C3%B5es%20e%20apoios%20atuais.>>. Acesso em: 4 jul. 2023, 19:30.
- RAMDOSS, S. et al. Use of Computer-Based Interventions to Teach Communication Skills to Children with Autism Spectrum Disorders: A Systematic Review. *Journal of Behavioral Education*, v. 20, n. 1, p. 55–76, 2011.
- REICHOW, B. et al. Social Skills Groups for People Aged 6 to 21 with Autism Spectrum Disorders (ASD). *Cochrane Database of Systematic Reviews*, v. 7, p. cd008511, 2012.
- ROCHA, C. *Poder do Python.* [S.l.: s.n.]. Disponível em: <<https://www.dio.me/articles/poder-do-python>>. Acesso em: 18 out. 2023, 20:00.
- ROSA, B. T. et al. As Dificuldades No Diagnóstico Do Transtorno Do Espectro Autista (Tea) Associadas À Ausência De Exames Específicos E À Percepção Familiar Perante Sinais Do Transtorno. *Científica E Cultural Da Medicina*, p. 32, 2022.
- ROSA, M. *Fazenda 4.0 A Automação Empregada Na Agricultura De Precisão*, 2020.
- SCHÖNING, J.; HOOK, J.; BARTINDALE, T. et al. Building Interactive Multi-touch Surfaces. In: *Tabletops - Horizontal Interactive Displays*. Edição: Christian Müller-Tomfelde. London: Springer London, 2010. P. 27–49. DOI: 10.1007/978-1-84996-113-4_2. Disponível em: <https://doi.org/10.1007/978-1-84996-113-4_2>.

- SOUZA AKAHOSI FERNANDES, A. D.; POLLI, L. M.; AGNELLI MARTINEZ, L. B. Características Psicomotoras e Sensoriais de crianças com Transtorno do Espectro Autista (TEA) em atendimento terapêutico ocupacional. *Revista Chilena de Terapia Ocupacional*, v. 22, n. 2, p. 137, dez. 2021. Disponível em: <<https://revistas.uchile.cl/index.php/RT0/article/view/57404>>. Acesso em: 16 set. 2022.
- TIAN, J.; GAO, X.; YANG, L. Repetitive Restricted Behaviors in Autism Spectrum Disorder: From Mechanism to Development of Therapeutics. *Frontiers in Neuroscience*, v. 16, p. 780407, mar. 2022. DOI: 10.3389/fnins.2022.780407.
- TREFFERT, D. A. Savant Syndrome: Realities, Myths, and Misconceptions. *Journal of Autism and Developmental Disorders*, v. 44, n. 3, p. 564–571, 2014.
- VUALA, E. A.; SOUSA, R. R. A. D.; CARTH, J. L. Automatização E Seu Impacto No Mercado De Trabalho.
- WEB, G. *Páginas web dinâmicas com JavaScript*. [S.l.: s.n.]. Disponível em: <<https://www.guiaweb.dev.br/7.-paginas-web-dinamicas-com-javascript>>. Acesso em: 4 ago. 2023, 19:00.
- WIKIPÉDIA. *Raspberry Pi*. [S.l.: s.n.]. Disponível em: <https://pt.wikipedia.org/wiki/Raspberry_Pi>. Acesso em: 17 out. 2023, 20:49.
- WORLD HEALTH ORGANIZATION. *World Report on Disability*. Geneva: [s.n.], 2011.

APÊNDICE A - COMPONENTES UTILIZADOS

Figura 18: *Raspberry Pi Zero W.*



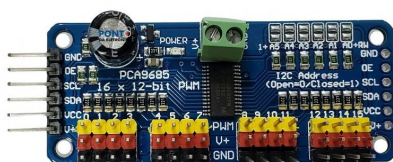
Fonte: do autor.

Figura 19: *Servo motor SG90.*



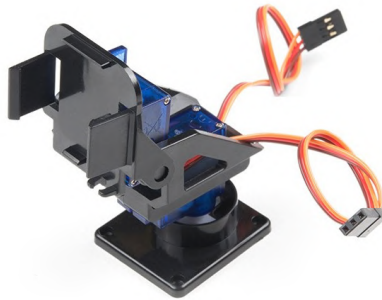
Fonte: do autor.

Figura 20: *Módulo PWM PCA9685.*



Fonte: do autor.

Figura 21: *Suporte Pan Tilt.*



Fonte: do autor.

Figura 22: *Módulo DTH11.*



Fonte: do autor.

Figura 23: *Cooler usado.*



Fonte: do autor.

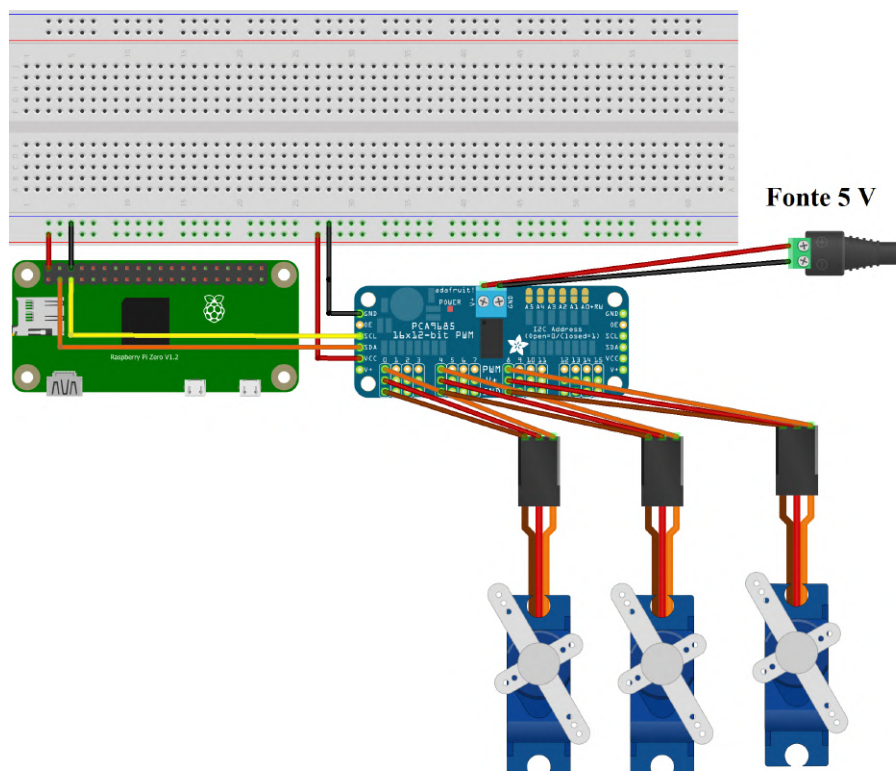
Figura 24: *Módulo relé usado.*



Fonte: do autor.

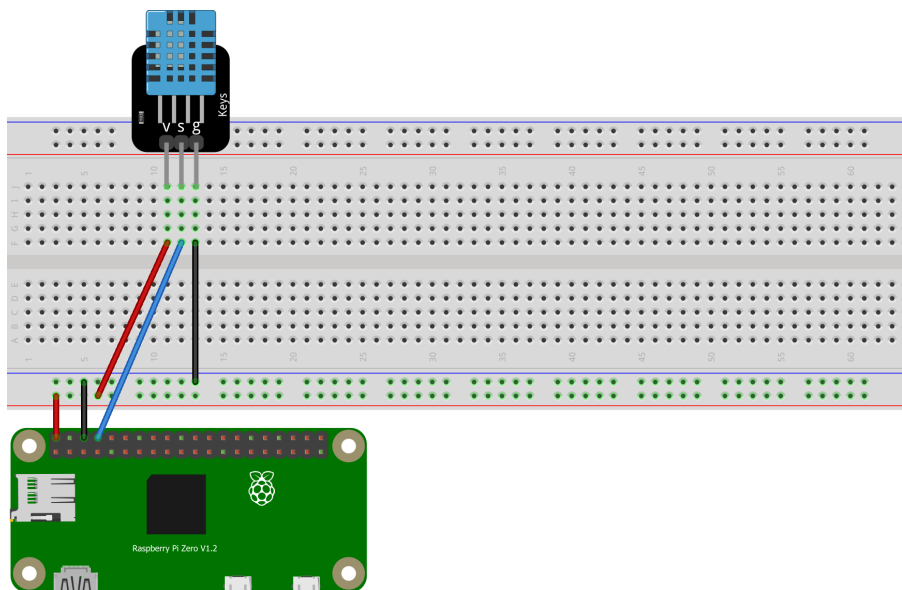
APÊNDICE B - ILUSTRAÇÃO DA MONTAGEM

Figura 25: Ilustração da montagem de componentes para automação da posição do espelho e da câmera infravermelho.



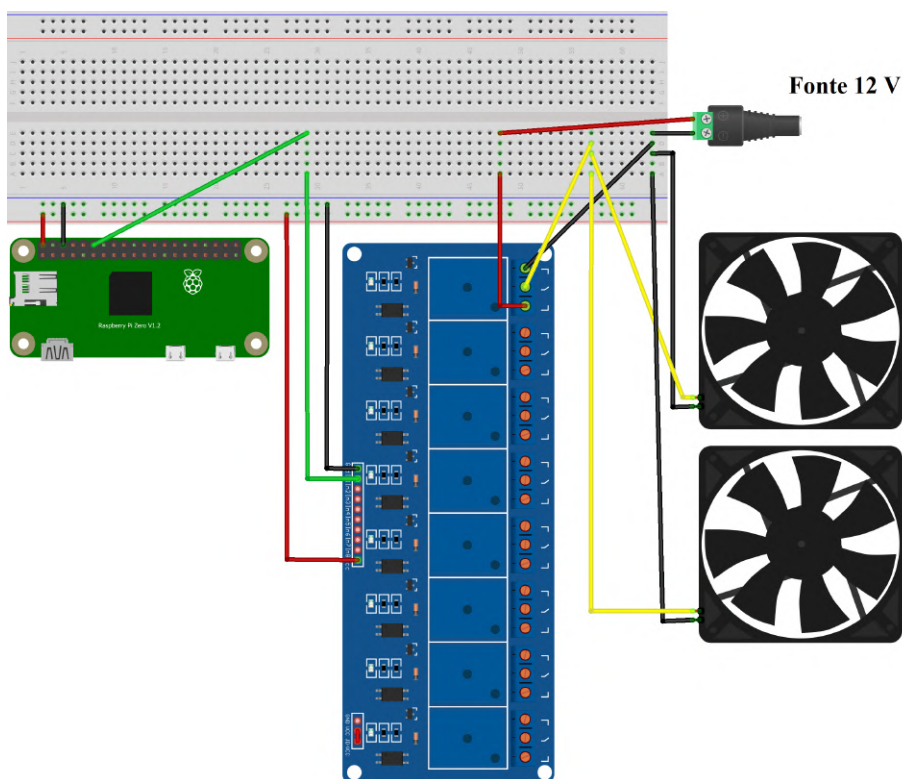
Fonte: do autor.

Figura 26: Ilustração da montagem do módulo DHT11 e do Raspberry Pi Zero W para automação do sistema de ventilação.



Fonte: do autor.

Figura 27: Ilustração da montagem dos coolers, do módulo relé de 8 canais, da fonte externa de 12 V e do Raspberry Pi Zero W para automação do sistema de ventilação.



Fonte: do autor.

APÊNDICE C - CÓDIGO ELABORADO EM PYTHON

Listing 5.1: Código em Python

```
# IMPORT FUNCTIONS
from PyQt5 import QtCore, QtGui, QtWidgets

import sys

import platform

from PySide2 import QtCore, QtGui, QtWidgets

from PySide2.QtCore import (QCoreApplication, QPropertyAnimation, QDate,
    QDateTime, QMetaObject, QObject, QPoint, QRect, QSize, QTime, QUrl, Qt,
    QEvent)

from PySide2.QtGui import (QBrush, QColor, QConicalGradient, QCursor, QFont,
    QFontDatabase, QIcon, QKeySequence, QLinearGradient, QPalette, QPainter,
    QPixmap, QRadialGradient)

from PySide2.QtWidgets import *

import RPi.GPIO as gpio

import Adafruit_DHT as dht

from PyQt5.QtCore import QTimer

from time import *

from adafruit_servokit import ServoKit

#VARIABLES GLOBAIS

usuario_cooler=0 #Usada para verificar acao do usuario para acionamento do
    cooler

x=0 #Usada para verificar acao do usuario na guia Menu
```



```

load1=18 #Pino usado para ligar e desligar cooler

gpio.setmode(gpio.BCM) #Definicao para modo BCM

gpio.setwarnings(False) #Desativacao de avisos

gpio.setup(load1,gpio.OUT) #Setando as portas de saida

kit = ServoKit(channels=16) #Setando a quantidade de canais disponiveis para
    uso pelo PCA9685

#CONFIGURAcAO DA ESTRUTURA DO PROGRAMA
class Ui_MainWindow(object):

    def setupUi(self, MainWindow):

        MainWindow.setObjectName("MainWindow")

        MainWindow.resize(700, 500)

        MainWindow.setMinimumSize(QtCore.QSize(500, 500))

        MainWindow.setStyleSheet("background-color:rgb(45,45,45);")

        self.centralwidget = QtWidgets.QWidget(MainWindow)

        self.centralwidget.setObjectName("centralwidget")

        self.verticalLayout = QtWidgets.QVBoxLayout(self.centralwidget)

        self.verticalLayout.setContentsMargins(0, 0, 0, 0)

        self.verticalLayout.setSpacing(0)

        self.verticalLayout.setObjectName("verticalLayout")

        self.Top_Bar = QtWidgets.QFrame(self.centralwidget)

        self.Top_Bar.setMaximumSize(QtCore.QSize(16777215, 40))

        self.Top_Bar.setStyleSheet("background-color:rgb(35,35,35);")

        self.Top_Bar setFrameShape(QtWidgets.QFrame.NoFrame)

        self.Top_Bar setFrameShadow(QtWidgets.QFrame.Raised)

        self.Top_Bar.setObjectName("Top_Bar")

        self.horizontalLayout = QtWidgets.QHBoxLayout(self.Top_Bar)

```

```

self.horizontalLayout.setContentsMargins(0, 0, 0, 0)

self.horizontalLayout.setSpacing(0)

self.horizontalLayout.setObjectName("horizontalLayout")

self.frame_toggle = QtWidgets.QFrame(self.Top_Bar)

self.frame_toggle.setMaximumSize(QtCore.QSize(70, 40))

self.frame_toggle.setStyleSheet("background-color:rgb(85,170,255);")

self.frame_toggle.setFrameShape(QtWidgets.QFrame.StyledPanel)

self.frame_toggle.setFrameShadow(QtWidgets.QFrame.Raised)

self.frame_toggle.setObjectName("frame_toggle")

self.verticalLayout_2 = QtWidgets.QVBoxLayout(self.frame_toggle)

self.verticalLayout_2.setContentsMargins(0, 0, 0, 0)

self.verticalLayout_2.setSpacing(0)

self.verticalLayout_2.setObjectName("verticalLayout_2")

self.Btn_Toggle = QtWidgets.QPushButton(self.frame_toggle)

    sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Expanding,
QtWidgets.QSizePolicy.Expanding)

    sizePolicy.setHorizontalStretch(0)

    sizePolicy.setVerticalStretch(0)

    sizePolicy.setHeightForWidth(self.Btn_Toggle.sizePolicy()).
hasHeightForWidth())

self.Btn_Toggle.setSizePolicy(sizePolicy)

font = QtGui.QFont()

font.setPointSize(10)

self.Btn_Toggle.setFont(font)

self.Btn_Toggle.setStyleSheet("QPushButton{\n"
"color:rgb(255,255,255);\n"
"background-color:rgb(169,169,169);\n"

```

```

"border: 0px solid; \n"

"} \n"

"QPushButton: hover { \n"

"background-color: rgb(85, 170, 255); \n"

"}")

    icon = QtGui.QIcon()

    icon.addPixmap(QtGui.QPixmap(":/icons/align-justify.svg"), QtGui.QIcon.
Normal, QtGui.QIcon.Off)

    self.Btn_Toggle.setIcon(icon)

    self.Btn_Toggle.setObjectName("Btn_Toggle")

    self.verticalLayout_2.addWidget(self.Btn_Toggle)

    self.horizontalLayout.addWidget(self.frame_toggle)

    self.frame_top = QtWidgets.QFrame(self.Top_Bar)

    self.frame_top.setFrameShape(QtWidgets.QFrame.StyledPanel)

    self.frame_top.setFrameShadow(QtWidgets.QFrame.Raised)

    self.frame_top.setObjectName("frame_top")

    self.horizontalLayout.addWidget(self.frame_top)

    self.verticalLayout.addWidget(self.Top_Bar)

    self.Content = QtWidgets.QFrame(self.centralwidget)

    self.Content.setFrameShape(QtWidgets.QFrame.NoFrame)

    self.Content.setFrameShadow(QtWidgets.QFrame.Raised)

    self.Content.setObjectName("Content")

    self.horizontalLayout_2 = QtWidgets.QHBoxLayout(self.Content)

    self.horizontalLayout_2.setContentsMargins(0, 0, 0, 0)

    self.horizontalLayout_2.setSpacing(0)

    self.horizontalLayout_2.setObjectName("horizontalLayout_2")

```

```

self.frame_left_menu = QtWidgets.QFrame(self.Content)

self.frame_left_menu.setMinimumSize(QtCore.QSize(70, 0))

self.frame_left_menu.setMaximumSize(QtCore.QSize(70, 16777215))

self.frame_left_menu.setStyleSheet("background-color:rgb(35,35,35);"
)

self.frame_left_menu setFrameShape(QtWidgets.QFrame.StyledPanel)

self.frame_left_menu setFrameShadow(QtWidgets.QFrame.Raised)

self.frame_left_menu.setObjectName("frame_left_menu")

self.horizontalLayout_3 = QtWidgets.QHBoxLayout(self.frame_left_menu)

self.horizontalLayout_3.setContentsMargins(0, 0, 0, 0)

self.horizontalLayout_3.setObjectName("horizontalLayout_3")

self.frame_top_menus = QtWidgets.QFrame(self.frame_left_menu)

self.frame_top_menus setFrameShape(QtWidgets.QFrame.NoFrame)

self.frame_top_menus setFrameShadow(QtWidgets.QFrame.Raised)

self.frame_top_menus.setObjectName("frame_top_menus")

self.verticalLayout_4 = QtWidgets.QVBoxLayout(self.frame_top_menus)

self.verticalLayout_4.setContentsMargins(0, 0, 0, 0)

self.verticalLayout_4.setSpacing(0)

self.verticalLayout_4.setObjectName("verticalLayout_4")

self.btn_page_1 = QtWidgets.QPushButton(self.frame_top_menus)

self.btn_page_1.setMinimumSize(QtCore.QSize(0, 40))

font = QtGui.QFont()

font.setPointSize(10)

self.btn_page_1.setFont(font)

self.btn_page_1.setStyleSheet("QPushButton{\n"
"color:rgb(255,255,255);\n"

```

```

"background-color:rgb(169,169,169);\n"

"border:0pxsolid;\n"

"}\n"

"QPushButton:hover{\n"

"background-color:rgb(85,170,255);\n"

"}")

    icon1 = QtGui.QIcon()

    icon1.addPixmap(QtGui.QPixmap(":/icons/thermometer.svg"), QtGui.QIcon.
Normal, QtGui.QIcon.Off)

    self.btn_page_1.setIcon(icon1)

    self.btn_page_1.setObjectName("btn_page_1")

    self.verticalLayout_4.addWidget(self.btn_page_1)

    self.btn_page_2 = QtWidgets.QPushButton(self.frame_top_menus)

    self.btn_page_2.setMinimumSize(QtCore.QSize(0, 40))

    font = QtGui.QFont()

    font.setPointSize(10)

    self.btn_page_2.setFont(font)

    self.btn_page_2.setStyleSheet("QPushButton{\n"

"color:rgb(255,255,255);\n"

"background-color:rgb(169,169,169);\n"

"border:0pxsolid;\n"

"}\n"

"QPushButton:hover{\n"

"background-color:rgb(85,170,255);\n"

"}")

    icon2 = QtGui.QIcon()

```

```

        icon2.addPixmap(QtGui.QPixmap(":/icons/move.svg"), QtGui.QIcon.Normal,
QtGui.QIcon.Off)

        self.btn_page_2.setIcon(icon2)

        self.btn_page_2.setObjectName("btn_page_2")

        self.verticalLayout_4.addWidget(self.btn_page_2)

        self.btn_page_3 = QtWidgets.QPushButton(self.frame_top_menus)

        self.btn_page_3.setMinimumSize(QtCore.QSize(0, 40))

        font = QtGui.QFont()

        font.setPointSize(10)

        self.btn_page_3.setFont(font)

        self.btn_page_3.setStyleSheet("QPushButton_{\n"
"color:rgb(255,255,255);\n"
"background-color:rgb(169,169,169);\n"
"border:0pxsolid;\n"
"}\n"
"QPushButton:hover_{\n"
"background-color:rgb(85,170,255);\n"
"}")

        icon3 = QtGui.QIcon()

        icon3.addPixmap(QtGui.QPixmap(":/icons/copy.svg"), QtGui.QIcon.Normal,
QtGui.QIcon.Off)

        self.btn_page_3.setIcon(icon3)

        self.btn_page_3.setObjectName("btn_page_3")

        self.verticalLayout_4.addWidget(self.btn_page_3)

        self.btn_page_ajuda = QtWidgets.QPushButton(self.frame_top_menus)

        self.btn_page_ajuda.setMinimumSize(QtCore.QSize(0, 40))

```

```

        font = QtGui.QFont()

        font.setPointSize(10)

        self.btn_page_ajuda.setFont(font)

        self.btn_page_ajuda.setStyleSheet("QPushButton_{\n"
"        color:rgb(255,255,255);\n"
"        background-color:rgb(169,169,169);\n"
"        border:0pxsolid;\n"
"}\n"
"QPushButton:hover_{\n"
"        background-color:rgb(85,170,255);\n"
"}")

        icon4 = QtGui.QIcon()

        icon4.addPixmap(QtGui.QPixmap(":/icons/help-circle.svg"), QtGui.QIcon.
Normal, QtGui.QIcon.Off)

        self.btn_page_ajuda.setIcon(icon4)

        self.btn_page_ajuda.setObjectName("btn_page_ajuda")

        self.verticalLayout_4.addWidget(self.btn_page_ajuda)

        self.btn_sair = QtWidgets.QPushButton(self.frame_top_menus)

        self.btn_sair.setMinimumSize(QtCore.QSize(0, 40))

        font = QtGui.QFont()

        font.setPointSize(10)

        self.btn_sair.setFont(font)

        self.btn_sair.setStyleSheet("QPushButton_{\n"
"        color:rgb(255,255,255);\n"
"        background-color:rgb(169,169,169);\n"
"        border:0pxsolid;\n"

```

```

"}\n"

"QPushButton: hover_{\n"

"background-color: rgb(85, 170, 255);\n"

"}")

    icon5 = QtGui.QIcon()

    icon5.addPixmap(QtGui.QPixmap(":/icons/x.svg"), QtGui.QIcon.Normal,
QtGui.QIcon.Off)

    self.btn_sair.setIcon(icon5)

    self.btn_sair.setObjectName("btn_sair")

    self.verticalLayout_4.addWidget(self.btn_sair)

    self.horizontalLayout_3.addWidget(self.frame_top_menus)

    self.horizontalLayout_2.addWidget(self.frame_left_menu)

    self.frame_pages = QtWidgets.QFrame(self.Content)

    self.frame_pages.setFrameShape(QtWidgets.QFrame.StyledPanel)

    self.frame_pages.setFrameShadow(QtWidgets.QFrame.Raised)

    self.frame_pages.setObjectName("frame_pages")

    self.verticalLayout_5 = QtWidgets.QVBoxLayout(self.frame_pages)

    self.verticalLayout_5.setObjectName("verticalLayout_5")

    self.stackedWidget = QtWidgets.QStackedWidget(self.frame_pages)

    self.stackedWidget.setObjectName("stackedWidget")

    self.page_1 = QtWidgets.QWidget()

    self.page_1.setEnabled(True)

    self.page_1.setObjectName("page_1")

    self.gridLayout = QtWidgets.QGridLayout(self.page_1)

    self.gridLayout.setObjectName("gridLayout")

    self.widget_10 = QtWidgets.QWidget(self.page_1)

```



```
self.widget_10.setObjectName("widget_10")

self.verticalLayout_12 = QtWidgets.QVBoxLayout(self.widget_10)

self.verticalLayout_12.setObjectName("verticalLayout_12")

self.lcdNumber_umidade = QtWidgets.QLCDNumber(self.widget_10)

self.lcdNumber_umidade.setStyleSheet("color:rgb(170,255,255);")

self.lcdNumber_umidade.setObjectName("lcdNumber_umidade")

self.verticalLayout_12.addWidget(self.lcdNumber_umidade)

self.gridLayout.addWidget(self.widget_10, 0, 1, 1, 1)

self.widget_11 = QtWidgets.QWidget(self.page_1)

self.widget_11.setObjectName("widget_11")

self.verticalLayout_17 = QtWidgets.QVBoxLayout(self.widget_11)

self.verticalLayout_17.setObjectName("verticalLayout_17")

self.label_8 = QtWidgets.QLabel(self.widget_11)

font = QtGui.QFont()

font.setPointSize(14)

self.label_8.setFont(font)

self.label_8.setStyleSheet("color:rgb(170,255,255);")

self.label_8.setObjectName("label_8")

self.verticalLayout_17.addWidget(self.label_8, 0, QtCore.Qt.AlignLeft|
QtCore.Qt.AlignVCenter)

self.gridLayout.addWidget(self.widget_11, 0, 2, 1, 1)

self.widget_12 = QtWidgets.QWidget(self.page_1)

self.widget_12.setStyleSheet("color:rgb(170,255,255);")

self.widget_12.setObjectName("widget_12")

self.verticalLayout_23 = QtWidgets.QVBoxLayout(self.widget_12)

self.verticalLayout_23.setObjectName("verticalLayout_23")
```

```
self.label_2 = QtWidgets.QLabel(self.widget_12)

font = QtGui.QFont()

font.setPointSize(12)

self.label_2.setFont(font)

self.label_2.setObjectName("label_2")

self.verticalLayout_23.addWidget(self.label_2, 0, QtCore.Qt.
AlignHCenter|QtCore.Qt.AlignVCenter)

self.gridLayout.addWidget(self.widget_12, 1, 0, 1, 1)

self.widget_13 = QtWidgets.QWidget(self.page_1)

self.widget_13.setObjectName("widget_13")

self.verticalLayout_15 = QtWidgets.QVBoxLayout(self.widget_13)

self.verticalLayout_15.setObjectName("verticalLayout_15")

self.lcdNumber_temperatura = QtWidgets.QLCDNumber(self.widget_13)

self.lcdNumber_temperatura.setStyleSheet("color:▯rgb(170,▯255,▯255);")

self.lcdNumber_temperatura.setObjectName("lcdNumber_temperatura")

self.verticalLayout_15.addWidget(self.lcdNumber_temperatura)

self.gridLayout.addWidget(self.widget_13, 1, 1, 1, 1)

self.widget_14 = QtWidgets.QWidget(self.page_1)

self.widget_14.setObjectName("widget_14")

self.verticalLayout_16 = QtWidgets.QVBoxLayout(self.widget_14)

self.verticalLayout_16.setObjectName("verticalLayout_16")

self.label_9 = QtWidgets.QLabel(self.widget_14)

font = QtGui.QFont()

font.setPointSize(14)

self.label_9.setFont(font)

self.label_9.setStyleSheet("color:▯rgb(170,▯255,▯255);")
```

```

self.label_9.setObjectName("label_9")

self.verticalLayout_16.addWidget(self.label_9, 0, QtCore.Qt.AlignLeft|
QtCore.Qt.AlignVCenter)

self.gridLayout.addWidget(self.widget_14, 1, 2, 1, 1)

self.widget_15 = QtWidgets.QWidget(self.page_1)

sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Minimum,
QtWidgets.QSizePolicy.Preferred)

sizePolicy.setHorizontalStretch(0)

sizePolicy.setVerticalStretch(0)

sizePolicy.setHeightForWidth(self.widget_15.sizePolicy().
hasHeightForWidth())

self.widget_15.setSizePolicy(sizePolicy)

self.widget_15.setObjectName("widget_15")

self.verticalLayout_8 = QtWidgets.QVBoxLayout(self.widget_15)

self.verticalLayout_8.setObjectName("verticalLayout_8")

self.Btn_cooler = QtWidgets.QPushButton(self.widget_15)

font = QtGui.QFont()

font.setPointSize(12)

self.Btn_cooler.setFont(font)

self.Btn_cooler.setStyleSheet("color:rgb(255,255,255);\n"
"background-color:rgb(169,169,169);\n"
"border:0pxsolid;")

self.Btn_cooler.setObjectName("Btn_cooler")

self.verticalLayout_8.addWidget(self.Btn_cooler)

self.gridLayout.addWidget(self.widget_15, 2, 0, 1, 3)

self.widget_6 = QtWidgets.QWidget(self.page_1)

self.widget_6.setObjectName("widget_6")

```

```
self.verticalLayout_11 = QtWidgets.QVBoxLayout(self.widget_6)

self.verticalLayout_11.setObjectName("verticalLayout_11")

self.label = QtWidgets.QLabel(self.widget_6)

font = QtGui.QFont()

font.setPointSize(12)

self.label.setFont(font)

self.label.setStyleSheet("color:▯rgb(170,▯255,▯255);")

self.label.setObjectName("label")

self.verticalLayout_11.addWidget(self.label, 0, QtCore.Qt.AlignHCenter|
QtCore.Qt.AlignVCenter)

self.gridLayout.addWidget(self.widget_6, 0, 0, 1, 1)

self.stackedWidget.addWidget(self.page_1)

self.page_2 = QtWidgets.QWidget()

self.page_2.setObjectName("page_2")

self.gridLayout_2 = QtWidgets.QGridLayout(self.page_2)

self.gridLayout_2.setObjectName("gridLayout_2")

self.widget_16 = QtWidgets.QWidget(self.page_2)

self.widget_16.setObjectName("widget_16")

self.verticalLayout_18 = QtWidgets.QVBoxLayout(self.widget_16)

self.verticalLayout_18.setObjectName("verticalLayout_18")

self.lcdNumber_horizontal_atual = QtWidgets.QLCDNumber(self.widget_16)

self.lcdNumber_horizontal_atual.setStyleSheet("color:▯rgb(170,▯255,▯
255);")

self.lcdNumber_horizontal_atual.setObjectName("
lcdNumber_horizontal_atual")

self.verticalLayout_18.addWidget(self.lcdNumber_horizontal_atual)

self.gridLayout_2.addWidget(self.widget_16, 2, 2, 1, 1)
```

```
self.widget_17 = QtWidgets.QWidget(self.page_2)

self.widget_17.setObjectName("widget_17")

self.verticalLayout_7 = QtWidgets.QVBoxLayout(self.widget_17)

self.verticalLayout_7.setObjectName("verticalLayout_7")

self.label_3 = QtWidgets.QLabel(self.widget_17)

font = QtGui.QFont()

font.setPointSize(12)

self.label_3.setFont(font)

self.label_3.setStyleSheet("color:rgb(170,255,255);")

self.label_3.setObjectName("label_3")

self.verticalLayout_7.addWidget(self.label_3, 0, QtCore.Qt.AlignHCenter
|QtCore.Qt.AlignVCenter)

self.gridLayout_2.addWidget(self.widget_17, 0, 2, 1, 2)

self.widget_9 = QtWidgets.QWidget(self.page_2)

sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Expanding,
QtWidgets.QSizePolicy.Preferred)

sizePolicy.setHorizontalStretch(0)

sizePolicy.setVerticalStretch(0)

sizePolicy.setHeightForWidth(self.widget_9.sizePolicy()).
hasHeightForWidth())

self.widget_9.setSizePolicy(sizePolicy)

self.widget_9.setObjectName("widget_9")

self.verticalLayout_19 = QtWidgets.QVBoxLayout(self.widget_9)

self.verticalLayout_19.setObjectName("verticalLayout_19")

self.lcdNumber_vertical_atual = QtWidgets.QLCDNumber(self.widget_9)

font = QtGui.QFont()

font.setPointSize(12)
```

```

self.lcdNumber_vertical_atual.setFont(font)

self.lcdNumber_vertical_atual.setStyleSheet("color:rgb(170,255,255);
")

self.lcdNumber_vertical_atual.setObjectName("lcdNumber_vertical_atual")

self.verticalLayout_19.addWidget(self.lcdNumber_vertical_atual)

self.gridLayout_2.addWidget(self.widget_9, 1, 2, 1, 1)

self.widget_7 = QtWidgets.QWidget(self.page_2)

self.widget_7.setObjectName("widget_7")

self.verticalLayout_14 = QtWidgets.QVBoxLayout(self.widget_7)

self.verticalLayout_14.setObjectName("verticalLayout_14")

self.spinBox_horizontal_desejada = QtWidgets.QSpinBox(self.widget_7)

font = QtGui.QFont()

font.setPointSize(12)

self.spinBox_horizontal_desejada.setFont(font)

self.spinBox_horizontal_desejada.setStyleSheet("color:rgb(255,255,255);\n"
"background-color:rgb(169,169,169);\n"
"border:0pxsolid;")

self.spinBox_horizontal_desejada.setMaximum(180)

self.spinBox_horizontal_desejada.setObjectName("spinBox_horizontal_desejada")

self.verticalLayout_14.addWidget(self.spinBox_horizontal_desejada)

self.gridLayout_2.addWidget(self.widget_7, 2, 1, 1, 1)

self.widget_5 = QtWidgets.QWidget(self.page_2)

self.widget_5.setObjectName("widget_5")

self.verticalLayout_13 = QtWidgets.QVBoxLayout(self.widget_5)

self.verticalLayout_13.setObjectName("verticalLayout_13")

```

```

self.spinBox_vertical_desejada = QtWidgets.QSpinBox(self.widget_5)

font = QtGui.QFont()

font.setPointSize(12)

self.spinBox_vertical_desejada.setFont(font)

self.spinBox_vertical_desejada.setStyleSheet("color: _rgb(255, _255, _255)
; \n"
"background-color: _rgb(169, _169, _169); \n"
"border: _0px _solid;")

self.spinBox_vertical_desejada.setMaximum(180)

self.spinBox_vertical_desejada.setObjectName("spinBox_vertical_desejada
")

self.verticalLayout_13.addWidget(self.spinBox_vertical_desejada)

self.gridLayout_2.addWidget(self.widget_5, 1, 1, 1, 1)

self.widget_4 = QtWidgets.QWidget(self.page_2)

self.widget_4.setObjectName("widget_4")

self.verticalLayout_6 = QtWidgets.QVBoxLayout(self.widget_4)

self.verticalLayout_6.setObjectName("verticalLayout_6")

self.label_4 = QtWidgets.QLabel(self.widget_4)

font = QtGui.QFont()

font.setPointSize(12)

self.label_4.setFont(font)

self.label_4.setStyleSheet("color: _rgb(170, _255, _255);")

self.label_4.setObjectName("label_4")

self.verticalLayout_6.addWidget(self.label_4)

self.gridLayout_2.addWidget(self.widget_4, 0, 1, 1, 1)

self.widget_3 = QtWidgets.QWidget(self.page_2)

self.widget_3.setObjectName("widget_3")

```

```
self.verticalLayout_3 = QtWidgets.QVBoxLayout(self.widget_3)

self.verticalLayout_3.setObjectName("verticalLayout_3")

self.label_7 = QtWidgets.QLabel(self.widget_3)

font = QtGui.QFont()

font.setPointSize(12)

self.label_7.setFont(font)

self.label_7.setStyleSheet("color: rgb(170, 255, 255);")

self.label_7.setObjectName("label_7")

self.verticalLayout_3.addWidget(self.label_7, 0, QtCore.Qt.AlignHCenter
)

self.gridLayout_2.addWidget(self.widget_3, 2, 0, 1, 1)

self.widget_2 = QtWidgets.QWidget(self.page_2)

self.widget_2.setObjectName("widget_2")

self.verticalLayout_22 = QtWidgets.QVBoxLayout(self.widget_2)

self.verticalLayout_22.setObjectName("verticalLayout_22")

self.label_6 = QtWidgets.QLabel(self.widget_2)

font = QtGui.QFont()

font.setPointSize(12)

self.label_6.setFont(font)

self.label_6.setStyleSheet("color: rgb(170, 255, 255);")

self.label_6.setObjectName("label_6")

self.verticalLayout_22.addWidget(self.label_6, 0, QtCore.Qt.
AlignHCenter)

self.gridLayout_2.addWidget(self.widget_2, 1, 0, 1, 1)

self.widget_19 = QtWidgets.QWidget(self.page_2)

font = QtGui.QFont()
```



```
font.setPointSize(12)

self.widget_19.setFont(font)

self.widget_19.setStyleSheet("color:rgb(170,255,255);")

self.widget_19.setObjectName("widget_19")

self.verticalLayout_20 = QtWidgets.QVBoxLayout(self.widget_19)

self.verticalLayout_20.setObjectName("verticalLayout_20")

self.label_11 = QtWidgets.QLabel(self.widget_19)

font = QtGui.QFont()

font.setPointSize(12)

self.label_11.setFont(font)

self.label_11.setObjectName("label_11")

self.verticalLayout_20.addWidget(self.label_11)

self.gridLayout_2.addWidget(self.widget_19, 2, 3, 1, 1)

self.widget_18 = QtWidgets.QWidget(self.page_2)

self.widget_18.setStyleSheet("color:rgb(170,255,255);")

self.widget_18.setObjectName("widget_18")

self.verticalLayout_21 = QtWidgets.QVBoxLayout(self.widget_18)

self.verticalLayout_21.setObjectName("verticalLayout_21")

self.label_10 = QtWidgets.QLabel(self.widget_18)

font = QtGui.QFont()

font.setPointSize(12)

self.label_10.setFont(font)

self.label_10.setObjectName("label_10")

self.verticalLayout_21.addWidget(self.label_10, 0, QtCore.Qt.
AlignHCenter)

self.gridLayout_2.addWidget(self.widget_18, 1, 3, 1, 1)
```

```
self.widget_XX = QtWidgets.QWidget(self.page_2)

self.widget_XX.setObjectName("widget_XX")

self.verticalLayout_9 = QtWidgets.QVBoxLayout(self.widget_XX)

self.verticalLayout_9.setObjectName("verticalLayout_9")

self.label_17 = QtWidgets.QLabel(self.widget_XX)

font = QtGui.QFont()

font.setPointSize(12)

self.label_17.setFont(font)

self.label_17.setStyleSheet("color:rgb(170,255,255);")

self.label_17.setObjectName("label_17")

self.verticalLayout_9.addWidget(self.label_17)

self.gridLayout_2.addWidget(self.widget_XX, 0, 0, 1, 1, QtCore.Qt.
AlignHCenter|QtCore.Qt.AlignVCenter)

self.stackedWidget.addWidget(self.page_2)

self.page_3 = QtWidgets.QWidget()

self.page_3.setObjectName("page_3")

self.gridLayout_3 = QtWidgets.QGridLayout(self.page_3)

self.gridLayout_3.setObjectName("gridLayout_3")

self.widget_24 = QtWidgets.QWidget(self.page_3)

font = QtGui.QFont()

font.setPointSize(12)

self.widget_24.setFont(font)

self.widget_24.setObjectName("widget_24")

self.verticalLayout_26 = QtWidgets.QVBoxLayout(self.widget_24)

self.verticalLayout_26.setObjectName("verticalLayout_26")

self.spinBox_espelho_desejada = QtWidgets.QSpinBox(self.widget_24)
```

```

        sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Minimum,
QtWidgets.QSizePolicy.Fixed)

        sizePolicy.setHorizontalStretch(0)

        sizePolicy.setVerticalStretch(0)

        sizePolicy.setHeightForWidth(self.spinBox_espelho_desejada.sizePolicy()
.hasHeightForWidth())

        self.spinBox_espelho_desejada.setSizePolicy(sizePolicy)

        font = QtGui.QFont()

        font.setPointSize(12)

        self.spinBox_espelho_desejada.setFont(font)

        self.spinBox_espelho_desejada.setStyleSheet("color:rgb(255,255,255)
;\n"
"background-color:rgb(169,169,169);\n"
"border:0pxsolid;")

        self.spinBox_espelho_desejada.setMaximum(180)

        self.spinBox_espelho_desejada.setObjectName("spinBox_espelho_desejada")

        self.verticalLayout_26.addWidget(self.spinBox_espelho_desejada)

        self.gridLayout_3.addWidget(self.widget_24, 1, 1, 1, 1)

        self.widget_8 = QtWidgets.QWidget(self.page_3)

        self.widget_8.setObjectName("widget_8")

        self.gridLayout_3.addWidget(self.widget_8, 0, 0, 1, 1)

        self.widget_20 = QtWidgets.QWidget(self.page_3)

        font = QtGui.QFont()

        font.setPointSize(12)

        self.widget_20.setFont(font)

        self.widget_20.setObjectName("widget_20")

        self.verticalLayout_27 = QtWidgets.QVBoxLayout(self.widget_20)

```

```
self.verticalLayout_27.setObjectName("verticalLayout_27")

self.label_13 = QtWidgets.QLabel(self.widget_20)

font = QtGui.QFont()

font.setPointSize(12)

self.label_13.setFont(font)

self.label_13.setStyleSheet("color:rgb(170,255,255);")

self.label_13.setObjectName("label_13")

self.verticalLayout_27.addWidget(self.label_13)

self.gridLayout_3.addWidget(self.widget_20, 0, 1, 1, 1)

self.widget_21 = QtWidgets.QWidget(self.page_3)

sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Expanding,
QtWidgets.QSizePolicy.Preferred)

sizePolicy.setHorizontalStretch(0)

sizePolicy.setVerticalStretch(0)

sizePolicy.setHeightForWidth(self.widget_21.sizePolicy().
hasHeightForWidth())

self.widget_21.setSizePolicy(sizePolicy)

font = QtGui.QFont()

font.setPointSize(12)

self.widget_21.setFont(font)

self.widget_21.setObjectName("widget_21")

self.verticalLayout_28 = QtWidgets.QVBoxLayout(self.widget_21)

self.verticalLayout_28.setObjectName("verticalLayout_28")

self.label_14 = QtWidgets.QLabel(self.widget_21)

font = QtGui.QFont()

font.setPointSize(12)

self.label_14.setFont(font)
```

```
self.label_14.setStyleSheet("color:rgb(170,255,255);")

self.label_14.setObjectName("label_14")

self.verticalLayout_28.addWidget(self.label_14, 0, QtCore.Qt.
AlignHCenter)

self.gridLayout_3.addWidget(self.widget_21, 0, 2, 1, 2)

self.widget_23 = QtWidgets.QWidget(self.page_3)

font = QtGui.QFont()

font.setPointSize(12)

self.widget_23.setFont(font)

self.widget_23.setObjectName("widget_23")

self.verticalLayout_24 = QtWidgets.QVBoxLayout(self.widget_23)

self.verticalLayout_24.setObjectName("verticalLayout_24")

self.label_5 = QtWidgets.QLabel(self.widget_23)

font = QtGui.QFont()

font.setPointSize(12)

self.label_5.setFont(font)

self.label_5.setStyleSheet("color:rgb(170,255,255);")

self.label_5.setObjectName("label_5")

self.verticalLayout_24.addWidget(self.label_5)

self.gridLayout_3.addWidget(self.widget_23, 1, 0, 1, 1)

self.widget_26 = QtWidgets.QWidget(self.page_3)

font = QtGui.QFont()

font.setPointSize(12)

self.widget_26.setFont(font)

self.widget_26.setObjectName("widget_26")

self.verticalLayout_25 = QtWidgets.QVBoxLayout(self.widget_26)
```

```
self.verticalLayout_25.setObjectName("verticalLayout_25")

self.label_12 = QtWidgets.QLabel(self.widget_26)

font = QtGui.QFont()

font.setPointSize(12)

self.label_12.setFont(font)

self.label_12.setStyleSheet("color:▯rgb(170,▯255,▯255);")

self.label_12.setObjectName("label_12")

self.verticalLayout_25.addWidget(self.label_12)

self.gridLayout_3.addWidget(self.widget_26, 1, 3, 1, 1)

self.widget_25 = QtWidgets.QWidget(self.page_3)

sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Expanding,
QtWidgets.QSizePolicy.Preferred)

sizePolicy.setHorizontalStretch(0)

sizePolicy.setVerticalStretch(0)

sizePolicy.setHeightForWidth(self.widget_25.sizePolicy().
hasHeightForWidth())

self.widget_25.setSizePolicy(sizePolicy)

font = QtGui.QFont()

font.setPointSize(12)

self.widget_25.setFont(font)

self.widget_25.setObjectName("widget_25")

self.verticalLayout_29 = QtWidgets.QVBoxLayout(self.widget_25)

self.verticalLayout_29.setObjectName("verticalLayout_29")

self.lcdNumber_superior = QtWidgets.QLCDNumber(self.widget_25)

self.lcdNumber_superior.setStyleSheet("color:▯rgb(170,▯255,▯255);")

self.lcdNumber_superior.setProperty("value", 0.0)
```

```

self.lcdNumber_superior.setProperty("intValue", 0)

self.lcdNumber_superior.setObjectName("lcdNumber_superior")

self.verticalLayout_29.addWidget(self.lcdNumber_superior)

self.gridLayout_3.addWidget(self.widget_25, 1, 2, 1, 1)

self.widget_27 = QtWidgets.QWidget(self.page_3)

self.widget_27.setObjectName("widget_27")

self.verticalLayout_30 = QtWidgets.QVBoxLayout(self.widget_27)

self.verticalLayout_30.setObjectName("verticalLayout_30")

self.gridLayout_3.addWidget(self.widget_27, 2, 0, 1, 1)

self.stackedWidget.addWidget(self.page_3)

self.page_4 = QtWidgets.QWidget()

self.page_4.setObjectName("page_4")

self.gridLayout_4 = QtWidgets.QGridLayout(self.page_4)

self.gridLayout_4.setObjectName("gridLayout_4")

self.comboBox = QtWidgets.QComboBox(self.page_4)

font = QtGui.QFont()

font.setPointSize(12)

self.comboBox.setFont(font)

self.comboBox.setStyleSheet("color: \u25a1rgb(255, \u25a1255, \u25a1255); \n"
"background-color: \u25a1rgb(169, \u25a1169, \u25a1169); \n"
"border: \u25a10px \u25a1solid;")

self.comboBox.setObjectName("comboBox")

self.comboBox.addItem("")

self.comboBox.addItem("")

self.comboBox.addItem("")

self.comboBox.addItem("")

```

```

self.comboBox.addItem("")

self.comboBox.addItem("")

self.gridLayout_4.addWidget(self.comboBox, 1, 0, 1, 1)

self.label_16 = QtWidgets.QLabel(self.page_4)

self.label_16.setObjectName("label_16")

self.gridLayout_4.addWidget(self.label_16, 3, 0, 1, 1)

self.pushButton = QtWidgets.QPushButton(self.page_4)

font = QtGui.QFont()

font.setPointSize(12)

self.pushButton.setFont(font)

self.pushButton.setStyleSheet("color:rgb(255,255,255);\n"
"background-color:rgb(169,169,169);\n"
"border:0pxsolid;")

self.pushButton.setObjectName("pushButton")

self.gridLayout_4.addWidget(self.pushButton, 2, 0, 1, 1)

self.label_15 = QtWidgets.QLabel(self.page_4)

font = QtGui.QFont()

font.setPointSize(12)

self.label_15.setFont(font)

self.label_15.setStyleSheet("color:rgb(170,255,255);")

self.label_15.setObjectName("label_15")

self.gridLayout_4.addWidget(self.label_15, 0, 0, 1, 1, QtCore.Qt.
AlignHCenter)

self.stackedWidget.addWidget(self.page_4)

self.verticalLayout_5.addWidget(self.stackedWidget)

self.horizontalLayout_2.addWidget(self.frame_pages)

```



```

self.verticalLayout.addWidget(self.Content)

MainWindow.setCentralWidget(self.centralwidget)

self.retranslateUi(MainWindow)

self.stackedWidget.setCurrentIndex(0)

QtCore.QMetaObject.connectSlotsByName(MainWindow)

def retranslateUi(self, MainWindow):
    _translate = QtCore.QCoreApplication.translate
    MainWindow.setWindowTitle(_translate("MainWindow", "MainWindow"))
    self.Btn_Toggle.setText(_translate("MainWindow", ""))
    self.btn_page_1.setText(_translate("MainWindow", ""))
    self.btn_page_2.setText(_translate("MainWindow", ""))
    self.btn_page_3.setText(_translate("MainWindow", ""))
    self.btn_page_ajuda.setText(_translate("MainWindow", ""))
    self.btn_sair.setText(_translate("MainWindow", ""))
    self.label_8.setText(_translate("MainWindow", "%"))
    self.label_2.setText(_translate("MainWindow", "Temperatura"))
    self.label_9.setText(_translate("MainWindow", "°C"))
    self.Btn_cooler.setText(_translate("MainWindow", "Cooler_OFF"))
    self.label.setText(_translate("MainWindow", "Umidade"))
    self.label_3.setText(_translate("MainWindow", "Posicao_Atual"))
    self.spinBox_horizontal_desejada.setSuffix(_translate("MainWindow", "°
"))
    self.spinBox_vertical_desejada.setSuffix(_translate("MainWindow", "°
"))

```

```

self.label_4.setText(_translate("MainWindow", "Posicao_Desejada"))

self.label_7.setText(_translate("MainWindow", "Posicao_Horizontal"))

self.label_6.setText(_translate("MainWindow", "Posicao_Vertical"))

self.label_11.setText(_translate("MainWindow", "Ã°"))

self.label_10.setText(_translate("MainWindow", "Ã°"))

self.label_17.setText(_translate("MainWindow", "Infravermelho"))

self.spinBox_espelho_desejada.setSuffix(_translate("MainWindow", "Ã°"))

self.label_13.setText(_translate("MainWindow", "Posicao_Desejada"))

self.label_14.setText(_translate("MainWindow", "Posicao_Atual"))

self.label_5.setText(_translate("MainWindow", "Espelho"))
)

self.label_12.setText(_translate("MainWindow", "Ã°"))

self.label_16.setText(_translate("MainWindow", "<html><head/><body><p align=\"justify\"><span style=\"font-family:\"arial\",\"sans-serif\";font-size:12pt;color:#aaffff;\">Ao pressionar o guia \"Menu\", ocorre a modificacao do layout para possibilitar a inclusao ou exclusao dos nomes dos guias.</span></p></body></html>"))

self.label_16.setWordWrap(True)

self.pushButton.setText(_translate("MainWindow", "Habilitar Selecao"))

self.label_15.setText(_translate("MainWindow", "Escolha uma opcao e aperte no botao \"Selecionar\":"))

self.label_15.setText(_translate("MainWindow", "Opre por uma das opcoes disponiveis e acione o botao:"))

self.comboBox.setItemText(0, _translate("MainWindow", "Menu"))

self.comboBox.setItemText(1, _translate("MainWindow", "Sistema de Ventilacao"))

self.comboBox.setItemText(2, _translate("MainWindow", "Ajustar Infravermelho"))

self.comboBox.setItemText(3, _translate("MainWindow", "Ajustar Espelho"))
))

self.comboBox.setItemText(4, _translate("MainWindow", "Ajuda"))

```

```
self.comboBox.setItemText(5, _translate("MainWindow", "Sair"))

#MANIPULAcAO PELA AcaO DE USUaRIO E DE DADOS LIDOS

def inicial_toggle(): #Funcao para animacao

    UIFunctions.toggleMenu(self, 250, True)

    global x

    if x<1: #Verificacao para layout de guias

        self.Btn_Toggle.setText(_translate("MainWindow", "Menu"))

        self.btn_page_1.setText(_translate("MainWindow", "Sistema de Ventilacao"))

        self.btn_page_2.setText(_translate("MainWindow", "Ajustar Infravermelho"))

        self.btn_page_3.setText(_translate("MainWindow", "Ajustar Espelho"))

        self.btn_page_ajuda.setText(_translate("MainWindow", "Ajuda"))

        self.btn_sair.setText(_translate("MainWindow", "Sair"))

        x=1

    else:

        self.Btn_Toggle.setText(_translate("MainWindow", ""))

        self.btn_page_1.setText(_translate("MainWindow", ""))

        self.btn_page_2.setText(_translate("MainWindow", ""))

        self.btn_page_3.setText(_translate("MainWindow", ""))

        self.btn_page_ajuda.setText(_translate("MainWindow", ""))

        self.btn_sair.setText(_translate("MainWindow", ""))

        x=0
```

```

        self.Btn_Toggle.clicked.connect(inicial_toggle) #Verifica se o guia
Menu foi apertado

def btn_sair_acionado(self): #Funcao para encerrar o programa

    gpio.cleanup() #Limpa os dados

    exit() #Encerra o programa

self.btn_sair.clicked.connect(btn_sair_acionado) #Verifica se o guia
Sair foi apertado

def acionar_cooler(): #Funcao para manipular dado se o usuario apertou
o botao para acionar cooler

    global usuario_cooler

    if usuario_cooler==0:

        usuario_cooler=1

    else:

        usuario_cooler=0

self.Btn_cooler.clicked.connect(acionar_cooler) #Verifica se o botao
para acionar cooler foi apertado

def Vertical_Desejada_Mudou(): #Funcao para atuar no servo do
infravermelho (vertical)

    kit.servo[0].angle = self.spinBox_vertical_desejada.value() #Servo
recebe o valor em graus

    self.lcdNumber_vertical_atual.display (round (kit.servo[0].angle))
#Atualiza posicao atual

self.spinBox_vertical_desejada.valueChanged.connect(
Vertical_Desejada_Mudou) #Verifica se o valor de angulo mudou

def Horizontal_Desejada_Mudou(): #Funcao para atuar no servo do
infravermelho (horizontal)

    kit.servo[4].angle = self.spinBox_horizontal_desejada.value() #
Servo recebe o valor em graus

```

```

        self.lcdNumber_horizontal_atual.display (round (kit.servo[4].angle)
) #Atualiza posicao atual

        self.spinBox_horizontal_desejada.valueChanged.connect(
Horizontal_Desejada_Mudou) #Verifica se o valor de angulo mudou

        def Espelho_Mudou(): #Funcao para atuar no espelho

            kit.servo[8].angle = self.spinBox_espelho_desejada.value() #Servo
recebe o valor em graus

            self.lcdNumber_superior.display (round (kit.servo[8].angle)) #
Atualiza posicao atual

            self.spinBox_espelho_desejada.valueChanged.connect(Espelho_Mudou) #
Verifica se o valor de angulo mudou

        def Combo_Mudou(): #Funcao para alterar o exposto em ajuda

            text = self.comboBox.currentText() #Variavel recebe a string para
comparacao

            if text == 'Menu':

                self.label_16.setText(_translate("MainWindow", "<html><head
/><body><p align=\"justify\"><span style=\"font-family:\'arial\',\'sans-
serif\';font-size:12pt;color:#aaffff;\">Ao pressionar o guia \"Menu\",
ocorre a modificacao do layout para possibilitar a inclusao ou exclusao dos
nomes dos guias.</span></p></body></html>"))

            if text == 'Sistema de Ventilacao':

                self.label_16.setText(_translate("MainWindow", "<html><head
/><body><p align=\"justify\"><span style=\"font-family:\'arial\',\'sans-
serif\';font-size:12pt;color:#aaffff;\">No referido guia, torna-se
possivel a verificacao dos niveis de umidade e temperatura. No caso de uma
temperatura superior a 20°C, a ventilacao e automaticamente ativada. Para
temperaturas inferiores a 20°C, a opcao de ativar ou desativar a
ventilacao esta disponivel ao usuario por meio do botao \"Cooler OFF\" e
Cooler ON\", respectivamente.</span></p></body></html>"))

            if text == 'Ajustar Infravermelho':

                self.label_16.setText(_translate("MainWindow", "<html><head
/><body><p align=\"justify\"><span style=\"font-family:\'arial\',\'sans-
serif\';font-size:12pt;color:#aaffff;\">Neste guia especifico, ha a
possibilidade de realizar ajustes na posicao do sensor infravermelho, com
valores aceitaveis variando de 0° a 180°. A selecao e efetuada ao inserir
o valor no campo de posicao desejada. O campo \"Posicao Vertical\" esta

```

```
relacionado,Ã  _movimentacao,na,direcao,do,eixo,vertical, _ao,passo,que,o,
campo,\"Posicao,Horizontal\",_esta,relacionado,Ã  _movimentacao,na,direcao,do
,eixo,horizontal. _</span></p></body></html>"))
```

```
if text == 'AjustarEspelho':
```

```
self.label_16.setText(_translate("MainWindow", "<html><head
/><body><p_align=\"justify\"><span_style=\"_font-family:\'arial\','_sans-
serif\';_font-size:12pt;_color:#aaffff;_\">Neste,guia, _e, _viavel, _efetuar,
ajustes, _na, _posicao, _do, _espelho, _cujos, _valores, _permitidos, _variam, _de, _0Ãº, _a, _180
Ãº. _A, _selecao, _e, _realizada, _ao, _inserir, _o, _valor, _no, _campo, _de, _posicao, _desejada. _
</span></p></body></html>"))
```

```
if text == 'Ajuda':
```

```
self.label_16.setText(_translate("MainWindow", "<html><head
/><body><p_align=\"justify\"><span_style=\"_font-family:\'arial\','_sans-
serif\';_font-size:12pt;_color:#aaffff;_\">Neste,guia, _e, _possivel, _acessar,
informacoes, _adicionalis, _sobre, _o, _funcionamento, _do, _programa. _A, _selecao, _e, _feita
_ao, _optar, _pelo, _guia, _desejado, _e, _pressionar, _o, _botao, _\"Habilitar, _Selecao\". _</
span></p></body></html>"))
```

```
if text == 'Sair':
```

```
self.label_16.setText(_translate("MainWindow", "<html><head
/><body><p_align=\"justify\"><span_style=\"_font-family:\'arial\','_sans-
serif\';_font-size:12pt;_color:#aaffff;_\">Ao, _pressionar, _este, _guia, _o, _
programa, _e, _encerrado. _</span></p></body></html>"))
```

```
self.pushButton.clicked.connect(Combo_Mudou) #Verifica se o botao para
alterar o exposto em ajuda foi pressionado
```

```
def update_sensor_umidade_temperatura(self): #Funcao para leitura do sensor
de temperatura
```

```
global y
```

```
global usuario_cooler
```

```
_translate = QtCore.QCoreApplication.translate
```

```
umidade, temperatura = dht.read_retry(dht.DHT11, 14) #Recebe valor de
temperatura e umidade
```

```
self.lcdNumber_umidade.display(umidade) #Expoe no display o valor da
umidade
```

```
self.lcdNumber_temperatura.display(temperatura) #Expoe no display o
valor da temperatura
```

```

T=20 #Temperatura de comparacao para ligar/desligar ventilacao

if temperatura > T:

    gpio.output(load1,gpio.LOW) #Acionar rele e liga cooler

    self.Btn_cooler.setText(_translate("MainWindow", "Cooler_ON"))

    self.Btn_cooler.setStyleSheet("color:rgb(170,255,127);\n"
"background-color:rgb(85,0,255);\n"
"border:10pxsolid;")

    if (temperatura <= T and usuario_cooler == 0):

        gpio.output(load1,gpio.HIGH) #Cooler desligado

        self.Btn_cooler.setText(_translate("MainWindow", "Cooler_OFF"))

        self.Btn_cooler.setStyleSheet("color:rgb(255,255,255);\n"
"background-color:rgb(169,169,169);\n"
"border:0pxsolid;")

        if (temperatura <= T and usuario_cooler == 1):

            gpio.output(load1,gpio.LOW) #Acionar rele e liga cooler

            self.Btn_cooler.setText(_translate("MainWindow", "Cooler_ON"))

            self.Btn_cooler.setStyleSheet("color:rgb(170,255,127);\n"
"background-color:rgb(169,169,169);\n"
"border:10pxsolid;")

#CONFIGURAcAO DA ANIMAcAO
class UIFunctions(Ui_MainWindow):

    def toggleMenu(self, maxWidth, enable):

        if enable:

            width = self.frame_left_menu.width()

            maxExtend = maxWidth

            standard = 70

```

```
        if width == 70:

            widthExtended = maxExtend

        else:

            widthExtended = standard

        self.animation = QPropertyAnimation(self.frame_left_menu, b"
minimumWidth")

        self.animation.setDuration(400)

        self.animation.setStartValue(width)

        self.animation.setEndValue(widthExtended)

        self.animation.setEasingCurve(QtCore.QEasingCurve.InOutQuart)

        self.animation.start()
```

```
import sources_rc #Simbolos usados nos guias
```

```
#JANELA PRINCIPAL
```

```
class MainWindow(QMainWindow):
```

```
    def __init__(self):
```

```
        QMainWindow.__init__(self)
```

```
        self.ui = Ui_MainWindow()
```

```
        self.ui.setupUi(self)
```

```
        self.ui.update_sensor_umidade_temperatura()
```

```
        self.timer = QTimer()
```

```
        self.timer.timeout.connect(lambda: self.ui.
update_sensor_umidade_temperatura())
```

```
        self.timer.start(3000) #Valor do sensor e atualizado a cada 3 segundos
```

```
#MODIFICACAO DA PAGINA MOSTRADA MEDIANDE A ACAO DO USUARIO NA SELECAO
DO GUIA
```



```
        self.ui.btn_page_1.clicked.connect(lambda: self.ui.stackedWidget.  
setCurrentWidget(self.ui.page_1))  
  
        self.ui.btn_page_2.clicked.connect(lambda: self.ui.stackedWidget.  
setCurrentWidget(self.ui.page_2))  
  
        self.ui.btn_page_3.clicked.connect(lambda: self.ui.stackedWidget.  
setCurrentWidget(self.ui.page_3))  
  
        self.ui.btn_page_ajuda.clicked.connect(lambda: self.ui.stackedWidget.  
setCurrentWidget(self.ui.page_4))  
  
        self.show()  
  
#EXECUCAO  
if __name__ == "__main__":  
  
    import sys  
  
    app = QtWidgets.QApplication(sys.argv)  
  
    ui = Ui_MainWindow()  
  
    window = MainWindow()  
  
    sys.exit(app.exec_())
```

APÊNDICE D - INSTRUMENTO EDUCACIONAL CRIADO

Figura 28: *Capa do recurso educacional.*



Fonte: do autor.

Figura 29: *Três abordagens no recurso educacional.*



Fonte: do autor.

Figura 30: Opções da trilha “Pareamento”.



Fonte: do autor.

Figura 31: Primeira tela da trilha “Pareamento: Animais”.



Fonte: do autor.

Figura 32: Segunda tela da trilha “Pareamento: Animais”.



Fonte: do autor.

Figura 33: Terceira tela da trilha “Pareamento: Animais”.



Fonte: do autor.

Figura 34: Quarta tela da trilha “Pareamento: Animais”.



Fonte: do autor.

Figura 35: Quinta tela da trilha “Pareamento: Animais”.



Fonte: do autor.

Figura 36: Primeira tela da trilha “Pareamento: Esportes”.



Fonte: do autor.

Figura 37: Segunda tela da trilha “Pareamento: Esportes”.



Fonte: do autor.

Figura 38: Terceira tela da trilha “Pareamento: Esportes”.



Fonte: do autor.

Figura 39: Quarta tela da trilha “Pareamento: Esportes”.



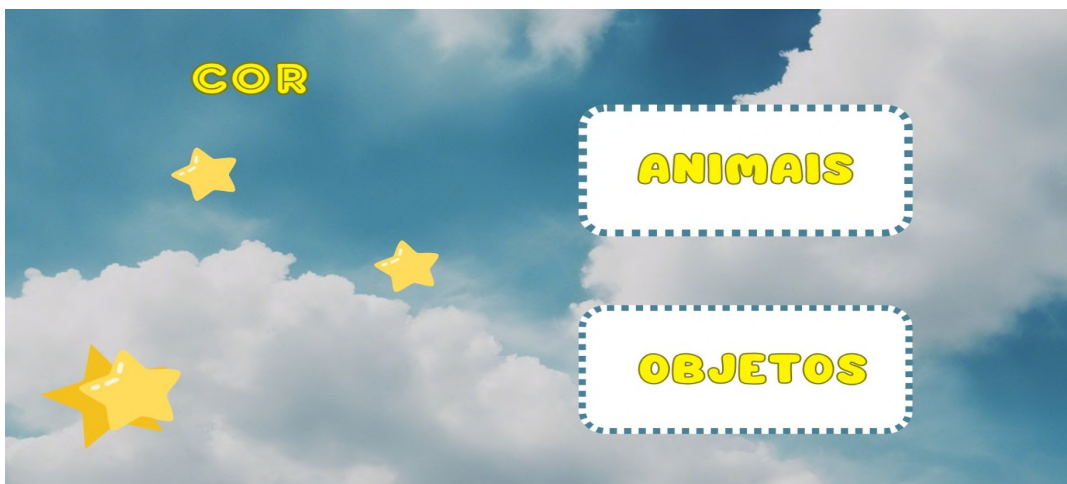
Fonte: do autor.

Figura 40: Quinta tela da trilha “Pareamento: Esportes”.



Fonte: do autor.

Figura 41: Opções da trilha “Cor”.



Fonte: do autor.

Figura 42: Primeira tela da trilha “Cor: Animais”.



Fonte: do autor.

Figura 43: Segunda tela da trilha “Cor: Animais”.



Fonte: do autor.

Figura 44: Terceira tela da trilha “Cor: Animais”.



Fonte: do autor.

Figura 45: Quarta tela da trilha “Cor: Animais”.



Fonte: do autor.

Figura 46: Quinta tela da trilha “Cor: Animais”.



Fonte: do autor.

Figura 47: Primeira tela da trilha “Cor: Objetos”.



Fonte: do autor.

Figura 48: Segunda tela da trilha “Cor: Objetos”.



Fonte: do autor.

Figura 49: Terceira tela da trilha “Cor: Objetos”.



Fonte: do autor.

Figura 50: Quarta tela da trilha “Cor: Objetos”.



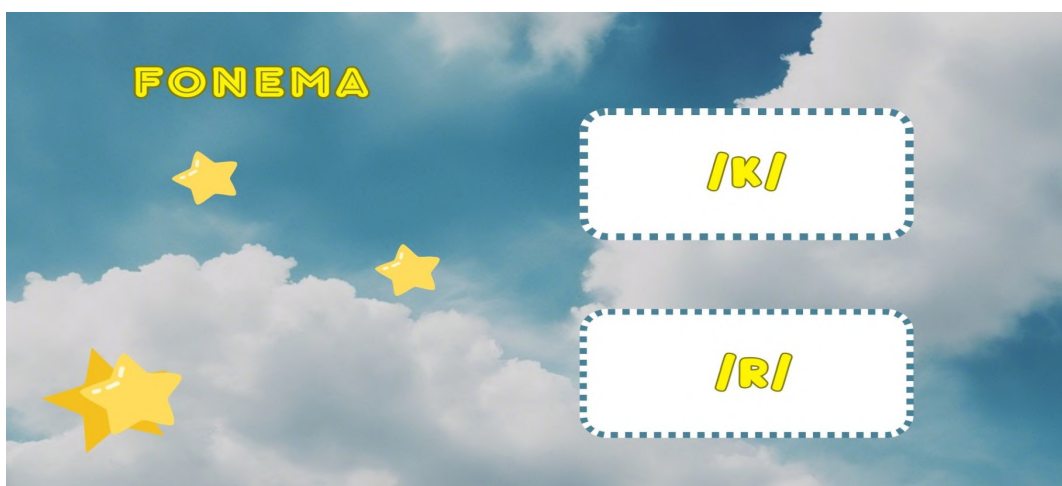
Fonte: do autor.

Figura 51: Quinta tela da trilha “Cor: Objetos”.



Fonte: do autor.

Figura 52: Opções da trilha “Fonema”.



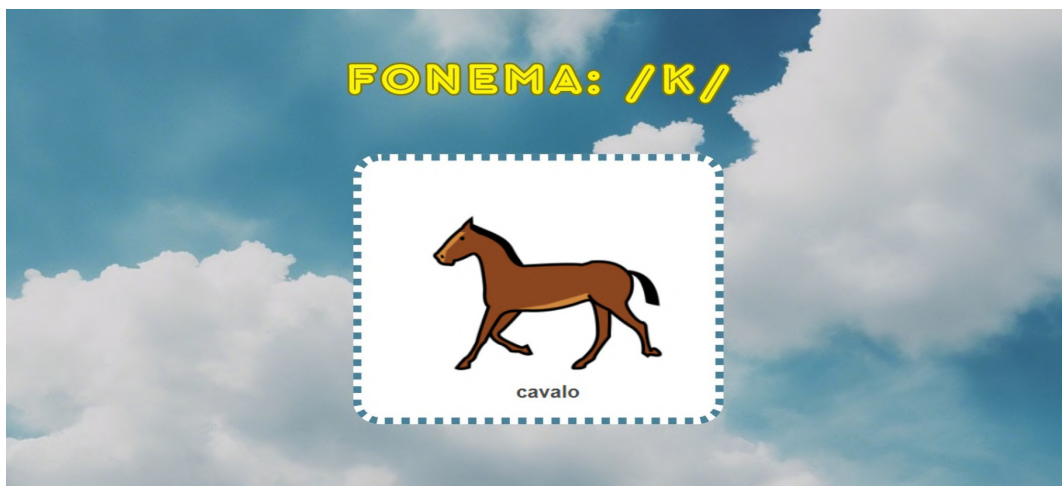
Fonte: do autor.

Figura 53: Primeira tela da apresentação da trilha “Fonema: /K/”.



Fonte: do autor.

Figura 54: Segunda tela da apresentação da trilha “Fonema: /K/”.



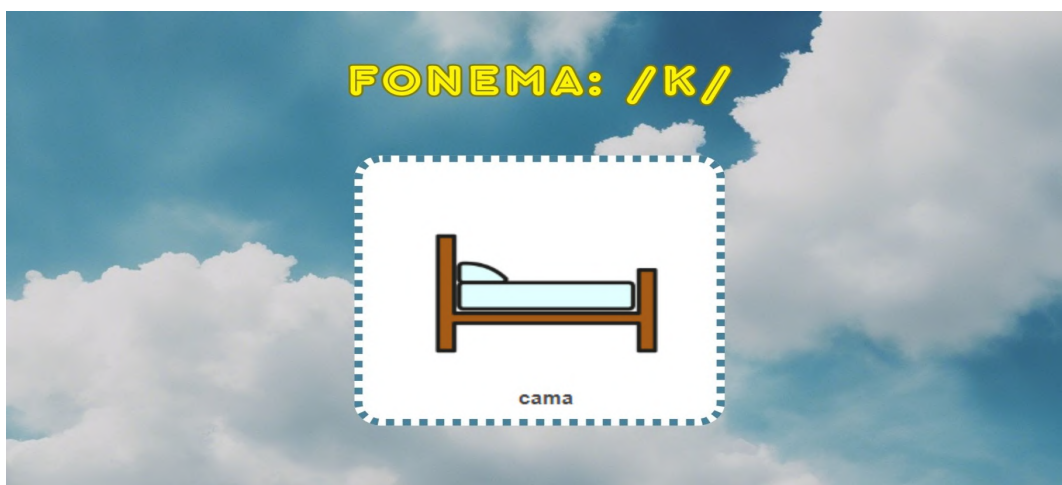
Fonte: do autor.

Figura 55: Terceira tela da apresentação da trilha “Fonema: /K/”.



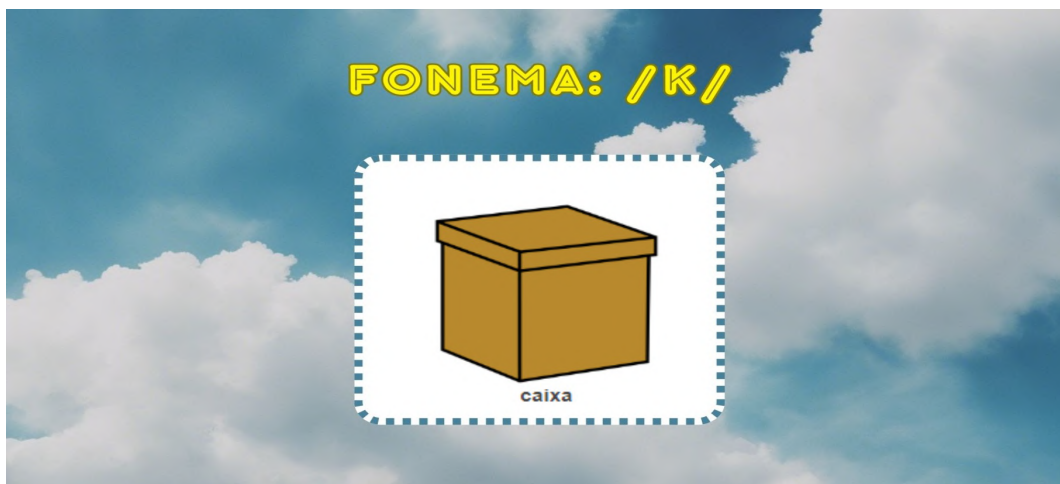
Fonte: do autor.

Figura 56: Quarta tela da apresentação da trilha “Fonema: /K/”.



Fonte: do autor.

Figura 57: Quinta tela da apresentação da trilha “Fonema: /K/”.



Fonte: do autor.

Figura 58: Primeira tela da interação da trilha “Fonema: /K/”.



Fonte: do autor.

Figura 59: Segunda tela da interação da trilha “Fonema: /K/”.



Fonte: do autor.

Figura 60: Terceira tela da interação da trilha “Fonema: /K/”.



Fonte: do autor.

Figura 61: Quarta tela da interação da trilha “Fonema: /K/”.



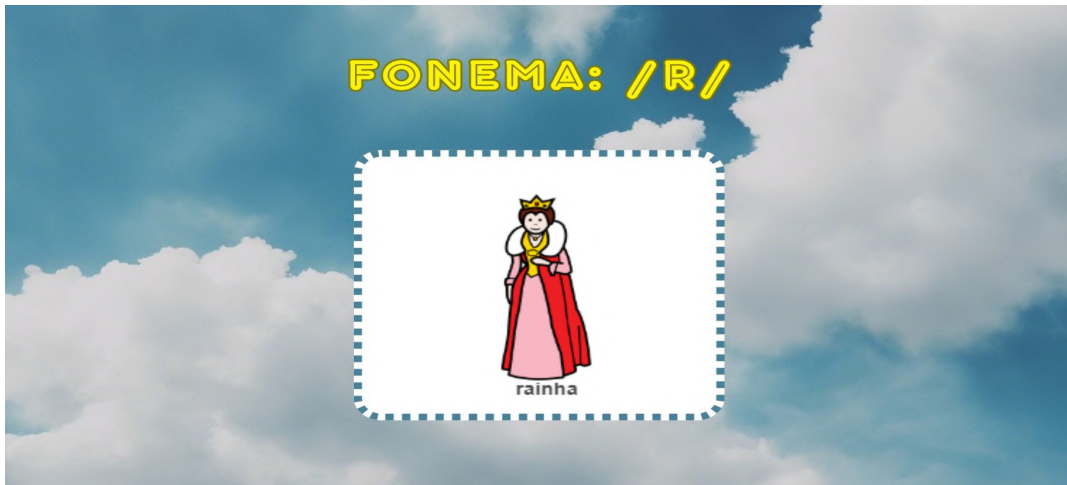
Fonte: do autor.

Figura 62: Quinta tela da interação da trilha “Fonema: /K/”.



Fonte: do autor.

Figura 63: Primeira tela da apresentação da trilha “Fonema: /R/”.



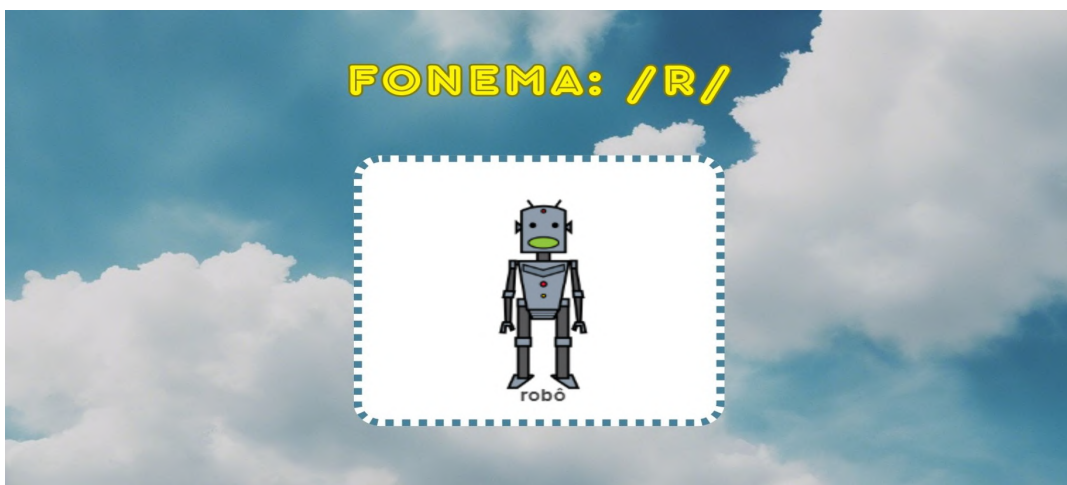
Fonte: do autor.

Figura 64: Segunda tela da apresentação da trilha “Fonema: /R/”.



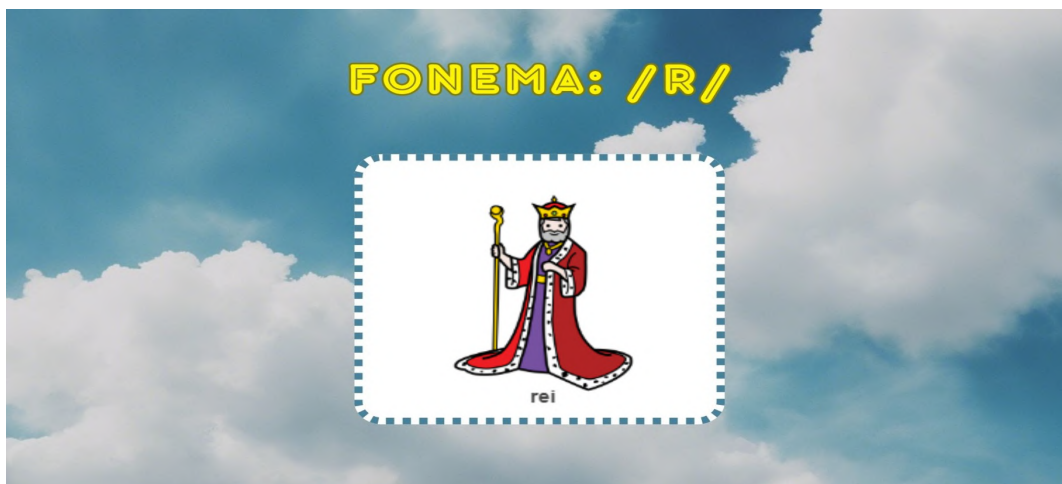
Fonte: do autor.

Figura 65: Terceira tela da apresentação da trilha “Fonema: /R/”.



Fonte: do autor.

Figura 66: Quarta tela da apresentação da trilha “Fonema: /R/”.



Fonte: do autor.

Figura 67: Quinta tela da apresentação da trilha “Fonema: /R/”.



Fonte: do autor.

Figura 68: Primeira tela da interação da trilha “Fonema: /R/”.



Fonte: do autor.

Figura 69: Segunda tela da interação da trilha “Fonema: /R/”.



Fonte: do autor.

Figura 70: Terceira tela da interação da trilha “Fonema: /R/”.



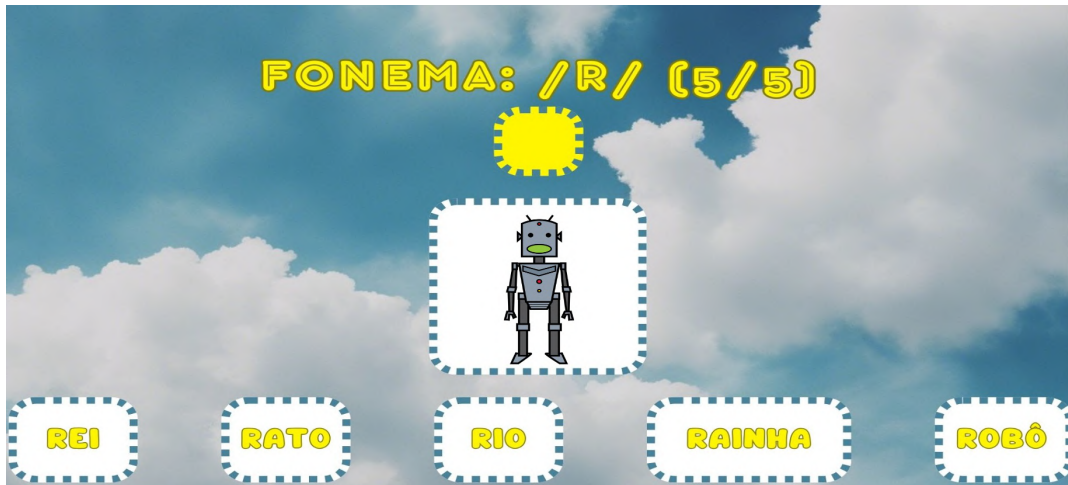
Fonte: do autor.

Figura 71: Quarta tela da interação da trilha “Fonema: /R/”.



Fonte: do autor.

Figura 72: Quinta tela da interação da trilha “Fonema: /R/”.



Fonte: do autor.

Figura 73: Tela de felicitações do recurso educacional.



Fonte: do autor.